

VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra informatiky

Zpracování a vizualizace provozních dat v korporátním prostředí

Data Processing and its Visualization within Corporate Environment

Zadání bakalářské práce

Student: **Ing. Jakub Kovář**

Studijní program: B2647 Informační a komunikační technologie

Studijní obor: 2612R025 Informatika a výpočetní technika

Téma: **Zpracování a vizualizace provozních dat v korporátním prostředí**
Data Processing and its Visualization within Corporate Environment

Jazyk vypracování: čeština

Zásady pro vypracování:

Cílem práce je ilustrovat dnešní možnosti integrace řešení pro zpracování, analýzu a vizualizaci dat v korporátním prostředí.

1. Popište základy problematiky Business Intelligence (BI).
2. Popište základní principy používané v této oblasti, a to s ohledem na jejich praktické provázání, tok dat a využívání datových zdrojů.
3. Srovnajte vybrané dostupné nástroje na vizualizaci a analytickou práci v rámci BI.
4. Navrhněte případovou studii, která pokryje komplexní proces práce s daty.
5. Realizujte případovou studii ve zvoleném prostředí.
6. Zhodnoťte možnosti dnešních BI nástrojů.

Seznam doporučené odborné literatury:


- [1] Rick Sherman: Business Intelligence Guidebook: From Data Integration to Analytics, Morgan Kaufmann, 2014, ISBN: 978-0124114616
- [2] Mike Morris: Power BI: A Complete Step-by-Step Guide for Beginners in Understanding Power BI, Independently published, 2019, ISBN: 978-1691641222
- [3] Joshua N. Milligan: Learning Tableau 2019: Tools for Business Intelligence, data prep, and visual analytics, Packt Publishing, 2019, ISBN: 978-1788839525

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.


Vedoucí bakalářské práce: **Ing. Michal Radecký, Ph.D.**

Datum zadání: 01.09.2019

Datum odevzdání: 30.04.2020



doc. Ing. Jan Platoš, Ph.D.
vedoucí katedry




prof. Ing. Pavel Brandštetter, CSc.
děkan fakulty

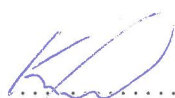
Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě 15. května 2020


.....

Souhlasím se zveřejněním této bakalářské práce dle požadavků čl. 26, odst. 9 Studijního a zkušebního řádu pro studium v bakalářských programech VŠB-TU Ostrava.

V Ostravě 15. května 2020

.....

Rád bych na tomto místě poděkoval panu Ing. Michalovi Radeckému, Ph.D., za veškeré odborné rady a ochotu při konzultacích této práce.

Abstrakt

Tato práce se zabývá problematikou zpracování a vizualizace dat v korporátním prostředí pomocí technik business intelligence. Zejména je zde řešen výběr moderního analytického a vizualizačního nástroje, vhodného pro vytvoření celofiremního komplexního reportingového řešení. Jádrem práce je porovnání tří nejvýznamnějších nástrojů pro analýzu a vizualizaci dat – Power BI, Tableau a QlikSense. Práce dále obsahuje i praktickou ukázkou extrakce, transformace a vizualizace dat za použití cloudové ETL platformy Keboola a vizualizačního nástroje Power BI.

Klíčová slova: Power BI; Tableau; QlikSense; business intelligence; vizualizace; Keboola; ETL; vizualizační nástroj; analýza dat

Abstract

This thesis focuses on data transformation and visualization in corporate environment using business intelligence techniques. In particular, the thesis focuses on choosing modern analytical and visualization tool which is suitable for creating a complex reporting solution across the whole company. The core of the thesis is comparison of the three most important data analysis and visualization tools - Power BI, Tableau and QlikSense. The thesis also contains a case study of extracting, transforming and visualizing data using cloud ETL platform Keboola and Power BI visualization tool.

Keywords: Power BI; Tableau; QlikSense; business intelligence; visualization; Keboola; ETL; visualization tool; data analysis

Obsah

Seznam použitých zkratk a symbolů	9
Seznam obrázků	10
Seznam tabulek	11
Seznam výpisů zdrojového kódu	12
1 Úvod	13
2 Problematika Business Intelligence	14
2.1 Proces vizualizace dat	14
2.2 Obecná architektura BI řešení	15
3 Srovnání nejdůležitějších nástrojů pro vizualizaci dat	20
3.1 Kritéria pro výběr nástrojů k porovnání	20
3.2 Porovnání v oblasti připojení k datovým zdrojům	21
3.3 Porovnání v oblasti rychlosti načtení datového zdroje	22
3.4 Porovnání v oblasti datového modelování	22
3.5 Porovnání práce s datovými typy	24
3.6 Porovnání možností rozšíření datového modelu pomocí vytváření tabulek, sloupců a metrik	25
3.7 Porovnání dostupnosti vývojového prostředí	27
3.8 Porovnání ovládání přístupu k datům	28
3.9 Porovnání tvorby vizualizací	28
3.10 Porovnání procesu tvorby dashboardů	30
3.11 Porovnání nákladů na uživatele	31
3.12 Další výhody a nevýhody jednotlivých nástrojů v bodech	32
3.13 Vyhovující datová základna pro moderní analytické nástroje	33
3.14 Shrnutí	33
4 Případová studie – vytížení vysokozdvížných vozíků	34
4.1 Počáteční situace	34
4.2 Cílový koncept	35
4.3 ETL proces a nahrání do cílové databáze	35
4.4 Vizualizace dat v Power BI	41
5 Shrnutí praktických výsledků	48

6 Závěr	49
Literatura	50
Přílohy	51
A Struktura složek v rámci přílohy	52
B Orientace v přiložené ukázce ETL Keboola	53

Seznam použitých zkratk a symbolů

API	– Application Programming Interface
BI	– Business Intelligence
CRM	– Customer Relationship Management
CSV	– Comma Separated Values
DB	– Databáze
DWH	– Data Warehouse
ETL	– Extract, Transform, Load
GPS	– Global Positioning System
VZV	– Vysokozdvížený vozík
XLS	– Excel Spreadsheet
XLSX	– Excel XML Format Spreadsheet

Seznam obrázků

1	Proces vizualizace dat, vlastní zpracování dle [6].	14
2	Bloková architektura BI systému, vlastní zpracování dle [7] a [4].	15
3	Rozdíl mezi ETL a ELT, překlad z [8].	17
4	Gartnerův magický kvadrant z [13].	20
5	Ukázka datového modelu v Power BI.	23
6	Bublinový datový model v Qlik Sense.	23
7	Klasický tabulkový datový model v Qlik Sense.	23
8	Ukázka datového modelu v Tableau.	24
9	Ukázka syntaxe vytvoření nového sloupce v Power BI.	26
10	Ukázka syntaxe vytvoření nové metriky v Qlik Sense.	26
11	Ukázka syntaxe vytvoření nové metriky v Tableau.	26
12	Ukázka jednoduché vizualizace v Power BI.	28
13	Ukázka jednoduché vizualizace v Tableau.	29
14	Ukázka jednoduché vizualizace v Qlik Sense.	30
15	Ukázka jednoduchého reportu s anonymizovanými názvy provozu v systému ISite.	34
16	Ukázka nastavení zasílání reportu v systému ISite.	36
17	Multiprojektová struktura v Keboola.	37
18	Finální tok dat zpracovávaných v rámci případové studie.	40
19	Ukázka hlavní strany dashboardu.	42
20	Ukázka detailního pohledu na nájezd vozíků v rámci závodu.	42
21	Datový model dashboardu v Power BI.	44
22	Nastavení podrobné analýzy v Power BI.	46
23	Nastavení synchronizace průřezů.	47

Seznam tabulek

1	Dostupnost datových konektorů.	21
2	Rychlost načtení dat z různých zdrojů.	22
3	Datové modelování	24
4	Vlastnosti rozšiřování datového modelu v jednotlivých nástrojích.	27
5	Možnost vývoje na různých operačních systémech.	27
6	Ceny za licenci a měsíc dle [14], [15] a [16]	31

Seznam výpisů zdrojového kódu

1	Řetězení procesorů v extraktoru Keboola	39
2	Vytvoření datové tabulky v jazyce DAX.	44
3	Vytvoření tabulky s odhadovými směnami řidičů v jazyce DAX.	45

1 Úvod

Ve firemním i občanském prostředí narůstá množství dat, která jsou k dispozici, a která dnes vytváří každý krok procesu. Dat je v rámci podniku a celého světa tolik, že vyznat se v nich začíná být čím dál tím větší problém. Již v knize Richarda Saula Wurmana *Information Anxiety* z roku 1989 se objevuje tvrzení, že v průměrném nedělním vydání novin *New York Times* je k dispozici více dat, než ke kterým měl v průběhu celého svého života přístup člověk žijící v období renesance. [1]

Pro úspěšné rozhodnutí manažerů ve firmách je důležité vyextrahovat z dat informace a za jejich pomoci se správně rozhodnout o daném problému. Proces, kdy se z faktu (data) stane informace, je popisován jako okamžik, kdy začneme pracovat s čísly a známe jejich hodnotu a význam. Pomocí informací, tak dokážeme učinit uvědomělé rozhodnutí. Posledním stupněm pochopení jsou pak znalosti, kdy informace vedou k predikci a ovlivní chování firmy či jednotlivce. [2]

Příkladem ke každému pojmu mohou být data narození zaměstnanců (data), z nichž je extrahována informace o věkové skladbě zaměstnanců v podniku a např. napříč odděleními. Akce/znalost vyplývající z toho zjištění je, že když je patrné, že v určitém oddělení pracuje mnoho lidí důchodového věku, tak k nim je přiřazen mladý člověk na zaučení. Transformace dat až ke znalosti je klíčová a čím dál tím těžší.

Vzhledem k rozsahu dat ale většinou nestačí podívat se na jednoduchou tabulku a na první pohled vidět problém. Pomyslná tabulka dnes totiž obsahuje milióny záznamů a pojmout je všechny najednou není v lidských silách. Proto se přistupuje k využití různých strategií pro analýzu a interpretaci dat, jako je statistika, dolování dat, strojové učení, umělá inteligence, deskriptivní a prediktivní modelování a mnoho dalších, z nichž jednou z nejdůležitějších je vizualizace – ať už vizualizace formou prezentace zjištění, nebo vizualizace dat k provedení exploratorní analýzy. [3] Což jsou oblasti, jimiž se zabývá multidisciplinární obor *Business Intelligence* (BI).

Cílem této práce je popsat problematiku BI, základní principy a techniky používané v BI řešeních, typický tok dat v podnikovém prostředí, a to vše s důrazem na vizualizaci dat. Hlavním úkolem je vybrat správný vizualizační nástroj pro podnik a popsat, co vše by měl umět nejen stran vizualizace, ale také modelování a analýzy dat, které od vizualizace samotné nelze jednoduše oddělit. Vybraný nástroj pak bude podroben detailnějšímu zkoumání a praktickému nasazení v případové studii.

2 Problematika Business Intelligence

Již v úvodu této práce bylo zmíněno, že Business Intelligence je multidisciplinární obor, který kombinuje podnikovou analytiku, statistické metody, metody strojového učení a deskriptivního modelování k tomu, aby manažerům firem pomohla řídit se ne na základě dojmů, ale na základě dat. Ve zjednodušeném, praktickém úhlu pohledu, má společnost správně nasazené BI nástroje tehdy, když se dokáže podívat na data vyprodukovaná společností i jejími zákazníky a použít je k tomu, aby společnost inovovala, odstraňovala neefektivní kroky vnitřních procesů a rychle se byla schopná adaptovat na výkyvy trhu. [3] [4]

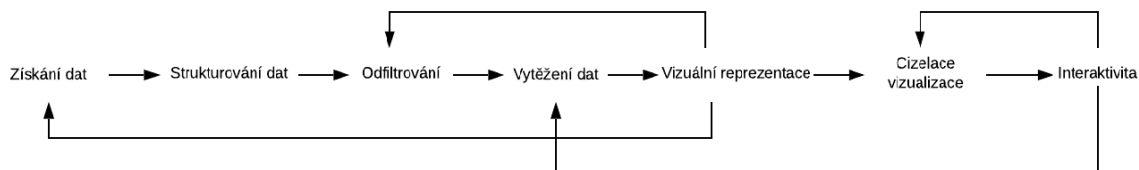
Dle [4] jsou na úkoly BI jsou typicky kladeny nároky ve smyslu sledování sledovaných podnikových ukazatelů a analyzování těchto ukazatelů (např. počet reklamací, tržba) z různých hledisek (dělení dle zákazníků, produktů). Moderní BI by mělo analyzovat vývoj těchto ukazatelů v čase a na různých úrovních detailu sledovaných hodnot.

Pro prezentaci zjištění vyplývajících z dat a jejich reprezentaci je pak využívána vizualizace. Vizualizace je jednou z nejlepších cest k pochopení rozsáhlých dat. Natahan Yau [5] nabádá: „Dejte čísla do vizuálních souvislostí a nechte čtenářův mozek, ať mezi nimi sám vypočítá určité zákonitosti. Takto je možné odhalit souvislosti, které byste za použití formálních statistických metod nikdy nenašli.“

I pouhá agregace dat a jejich vizualizace někdy umožní je proměnit v informace. Samozřejmě, data je nutné nejdříve získat, zpracovat a až poté je vizualizovat.

2.1 Proces vizualizace dat

Tento proces je popsán v [6], kde je odkázáno na fakt, že BI je oblast široce multidisciplinární a specialisté v tomto oboru musí znát základy programování, statistiky i vizuálního modelování. Jednotlivé kroky procesu pak navrhuje neizolovat, ale naopak spojovat tak, aby každý specialista znal všechny návaznosti. Jednoduché schéma procesu i možné interakce jednotlivých jeho kroků jsou znázorněny na Obrázku 1.



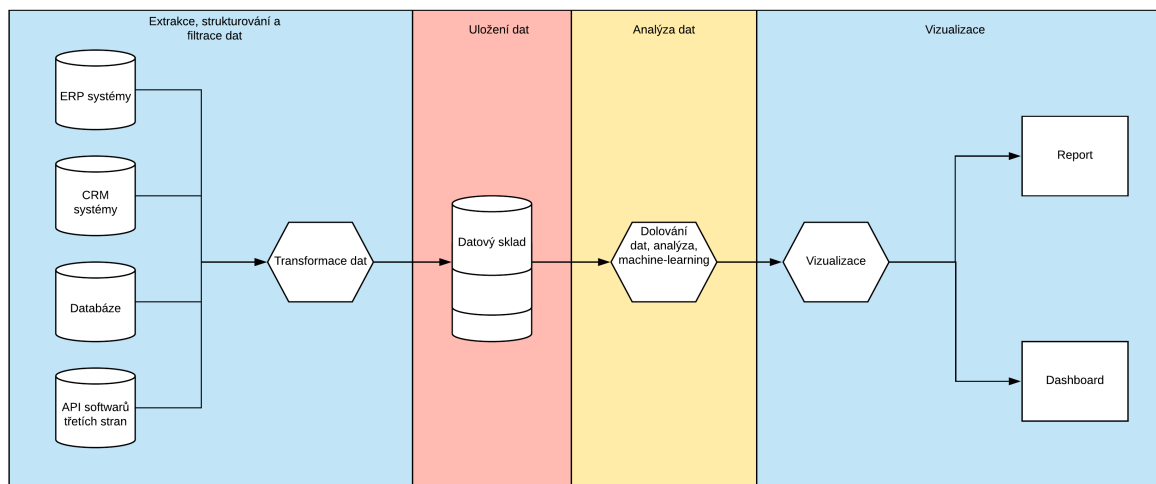
Obrázek 1: Proces vizualizace dat, vlastní zpracování dle [6].

- Získání dat – znamená většinou programátorskou úlohu napojení na zdrojový systém dat a stažení těchto dat do cílového systému, typicky reportingové databáze.

- Strukturování dat – kategorizace dat, systematizace a zapsání do databáze tak, aby každý sloupec tabulky měl svůj jednoznačný datový typ.
- Filtrace dat – využití jen těch dat, která mají pro vizualizaci smysl.
- Vytěžení dat – formulace základních zjištění, které lze získat – počítání průměrů, minim, maxim, odchylek a další.
- Vizualní reprezentace – rozhodnutí, jakou formou je vhodné reprezentovat data: zda jde o tabulku, jednoduchý sloupcový graf, ganttův diagram a jiné.
- Cizelace vizualizace – možnost zavedení různých hierarchií, aby se k datům dalo přistupovat v různém detailu, měnění barev pro lepší čitelnost grafu, podmíněné formátování atd.
- Interakce – nechat uživatele, aby byl dynamicky schopný měnit rozsah vizualizace a hledat v ní svá zjištění.

2.2 Obecná architektura BI řešení

Aby bylo možné v kapitole 2 popsany proces vizualizace provést, je v BI systémech nutné udržovat komponenty, které jednotlivé kroky tohoto procesu zajistí. Blokové schéma těchto komponent je znázorněno na Obrázku 2.



Obrázek 2: Bloková architektura BI systému, vlastní zpracování dle [7] a [4].

Různé zdroje ([7], [4], [8]) se shodují na tomto rozvětvení do čtyř vzájemně propojených subsystémů, které mají za úkol transformovat data v informaci a v nějaké podobě ji přinést koncovému uživateli. V rámci tohoto schématu jsou rozeznávány čtyři různé typy komponent:

- Komponenty pro extrakci, transformaci, čištění a nahrávání dat – komponenty určené pro extrakci dat ze zdrojových systémů, jejich transformaci a následné předání do komponentů pro ukládání dat.
- Komponenty pro ukládání dat – zajišťují ukládání, aktualizaci a správu dat.
- Komponenty pro analýzu dat – většinou technické zpracování dat, jejich dolování, dotazy do databázových komponent.
- Komponenty pro vizualizaci dat – konkrétní aplikace umožňující uživatelům konzumovat reporty.

V moderních BI systémech je možné, že se role jednotlivých komponent mohou prolínat a jeden software zastane roli více vzájemně sousedících oblastí.

2.2.1 Extrakce, transformace a čištění dat

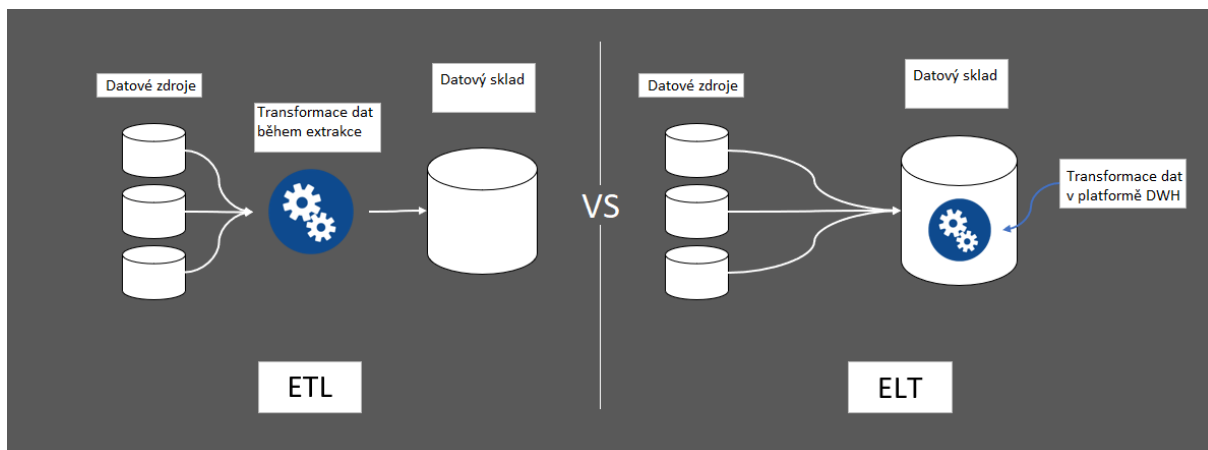
Při extrakci a transformaci dat je v prvním kroku důležité správně identifikovat zdroje dat [7] – pro většinu procesů je zdrojů dat více, a jde o zdroje dat interní (reálné prodeje výrobků zkoumané firmy) i externí (zakoupená data o daném tržním segmentu a jeho výkonu). Těmito zdroji dat mohou být ERP systémy, CRM systémy, různé databáze, API softwarů třetích stran a další.

Úkolem BI je v případě zdrojových systémů zajistit přístup ke správným a aktuálním datům. Jedním z nejdůležitějších rozhodnutí je to o granularitě dat [4]. Granularita dat znamená míru detailu dat – čím větší granularita, tím menší úvodní agregace dat. Největší granularita dat znamená např. v rámci faktury jít na jednotlivé položky faktury včetně detailního pohledu na zboží a jeho cenu, menší granularita pak znamená znát např. jen souhrnnou cenu za celou fakturu.

Jakmile je rozhodnuto o granularitě dat, je možné zajistit přístup k datům pro nástroje, jež data následně automatizovaně extrahují. Jedná se o proces nastavení tzv. datové pumpy, nebo-li ETL procesu. U něj jde o to, vybrat a získat (Extract) data ze zdrojových systémů, tato data následně transformovat do podoby vhodné pro reporting a vyčistit anomálie v datech (Transform), a následně je nahrát do datového skladu (Load). [4] Pro realizaci těchto kroků se často využívají skripty, nebo specializované nástroje.

S rozvojem cloudové architektury, a tím související větší výpočetní síly i zlevnění úložišť se často objevuje modernější varianta ETL, kterou je ELT [8].

Hlavním rozdílem je v tomto případě extrakce všech dat v co největší granularitě, jejich uložení do datového skladu již na vstupu a průběh transformací dat až přímo v něm, viz Obrázek 3. Výhodou tohoto přístupu je, že pokud se objeví nutnost využít dříve filtrovaná data nebo atributy, které se netransformovaly, není nutné data znovu extrahovat, ale pouze přenastavit transformaci.



Obrázek 3: Rozdíl mezi ETL a ELT, překlad z [8].

2.2.2 Uložení a prvotní analýza dat

V rámci ukládání dat je nejdůležitějším pojmem datový sklad (Data Warehouse – DWH), což je úložiště dat, které dle [4], [9] naplňuje následující charakteristiky:

- Je subjektivně orientované, což znamená, že data jsou rozdělena podle jednotlivých subjektů, např. všechny faktury jsou společně na jednom místě, a ne na různých místech dle míst vzniku (různé ERP systémy).
- Je integrované, tedy se do něj ukládají data z celého podniku.
- Je stálé, tedy uložená data nelze jednoduše napřímo měnit.
- Poslední podmínkou je časová rozlišenost, kdy je nutné, aby v DWH byla uložena informace o historii dat a načítaná data s sebou nesou informaci ohledně času pořízení a platnosti.

Obecným doporučením také je, pokud to kapacity dovolují, uchovávat data v datovém skladu s nejvyšší možnou granularitou (agregace dat je jednoduchá, jejich rozpad je přesně nemožný, lze ale odhadnout a alokovat).

Po uložení dat do DWH nastává proces, jež Sherman [7] označuje jako „data franchising“ – a tím je transformace dat do informací pro další reporting a analýzu. Data jsou v průběhu tohoto procesu znovu filtrována, reorganizována, agregována, zpracována a znovu uložena, tentokrát už s cílem konkrétního použití v konkrétní oblasti.

Po takovéto transformaci se data ukládají do tzv. datového tržiště, které obsahuje data pro využití v konkrétních analytických nebo vizualizačních aplikacích. Někteří autoři [4] již tento krok označují jako prvotní analýzu dat.

V rámci uložení do datového tržiště jsou data často remodelována tak, aby odpovídala standardům BI aplikací a kostek. Primárním cílem je data, pokud je to možné, rozdělit na faktová a dimenzionální a využít tak tzv. dimenzionálního modelování [10]. To znamená navržení datového

modelu tak, aby v rámci modelu bylo několik dimenzionálních tabulek s potřebnými kmenovými (master) daty a na ně napojenu ideálně jednu, nebo více tabulek faktů.

2.2.3 Typy datových tabulek

Tabulky faktů představují tabulky dat, jež obsahují sledované hodnoty ukazatelů identifikované skrze primární klíče. Tyto ukazatele jsou pak v rámci tabulky vázány na konkrétní master data – třeba na zákazníka, produkt. Existují dva speciální případy faktových dat dle [4]:

Transakční data se vztahují vždy ke konkrétním transakcím – průběhům procesu. Typickým představitelem transakčních dat je třeba kniha faktur, tabulka dodacích listů a další.

Periodicky snímkané tabulky faktů představují pravidelné časové otisky stavu určitých ukazatelů. V praxi jde například o počty palet na skladě, které se přenášejí každý den ve stejnou hodinu a v rámci roku tvoří ucelenou časovou řadu stavových veličin.

Master data pak představují jakési tabulky vlastností dat: popisné a doplňující informace důležité pro další prozkoumání transakčních dat. Pro příklad jde třeba o konkrétní adresu zákazníka, kontaktní osoby, GPS lokaci provozovny. . .

2.2.4 Speciální schémata dimenzionálního modelování

Speciálními případy dimenzionálního modelování jsou hvězdicový a vločkový model [11], [9]. Hvězdicový model je více denormalizovaný a v rámci jednotlivých dimenzí se vyskytují částečně redundantní data (např. Id kategorie a Název kategorie, Id podkategorie a název podkategorie).

V rámci vločkového modelu se dimenzionální tabulky mezi sebou dále řetěží s kardinalitou 1:N. Údržba modelu je jednodušší, nicméně vnitřní joiny mezi tabulkami jsou složitější.

2.2.5 Analýza dat

Jakmile jsou data připravena v odpovídajícím formátu, je možné je začít analyzovat. Jako analýzu lze chápat [10] i jednoduchou agregaci, při níž jsou vypočteny sumy různých částek, průměry jako je průměrná výše objednávky, nebo pokročilejší, jako je identifikování trendů a statistická analýza korelace.

V mnoha případech je nutné pro analýzu dat využít programovacích jazyků jako je Python, R a SQL, nebo využití machine learning a umělé inteligence. Mezi starší zástupce těchto softwarů patří i Excel, Google Tabulky, statistický software SPSS a další.

Analýza může probíhat buď v samostatném programu, nebo může probíhat na platformě datového skladu, či naopak být vnořena do vizualizačního nástroje. Není výjimkou, když složitější programy jsou součástí řešení DWH, a jednodušší analytické operace nad již zpracovanými daty jsou součástí aplikací pro vizualizaci.

2.2.6 Vizualizace dat

Poslední důležitou komponentou BI systému jsou jednotlivé vizualizace, reporty, dashboardy, listy Excelu, obrázky na nástěnkách. Jde o finální zpracování dat, prezentace provedené analýzy. Lidé obecně vnímají vizuálně, a proto je zobrazení formou správných grafů a tabulek klíčové. Ne nadarmo říká staré přísloví, že „obrázek vydá za tisíc slov.“

Vizualizace umožňuje jedním grafem agregovat, stovky, tisíce i desetitisíce datových bodů a poskytuje srozumitelný způsob, jak takové množství dat vstřebat. [10] Vizualizace umožňuje, aby její autor mohl vyprávět „příběh“ dat: například ten, že za poklesem finančních výsledků v daném segmentu stojí snížení průměrné výše objednávky, za níž stojí všeobecně nižší prodeje konkrétního produktu v konkrétním kraji.

V rámci vizualizací je důležité sledovat a zachycovat jisté obecně pochopitelné vzorce (např. sezónnost v časových řadách, obecně vývoj nějaké veličiny) a zvýraznit vztahy [5] mezi jednotlivými grafy. Když je tak patrný klesající zisk, je vhodné doplnit graf druhým, kde lze vidět vývoj nákladů a příjmů, abych bylo patrné, co a do jaké míry se na klesání zisku podílí.

Dle [12] je samotná vizualizace také součástí analýzy, protože vizuálním prozkoumáním dat (tzv. exploratorní analýzou) lze lépe pochopit trendy, na první pohled identifikovat odlehle hodnoty a prozkoumat vztahy mezi daty.

Pro vizualizaci a analýzu dat se v současné době začínají čím dál častěji používat aplikace třetích stran, které umožňují integraci těchto dvou vrstev BI systémů.

Pro posun k moderní vizualizaci je nutné tyto nástroje začlenit do podnikových struktur a využít je k novému pohledu na data – starší nástroje často umožňovaly jen pohled skrze jednoduché grafy anebo nejčastěji tabulky, které nebyly interaktivní, a které při dnešním rozsahu dat nestačí pro efektivní transformaci dat v informace.

Porovnání moderních nástrojů je provedeno v třetí kapitole a při výběru je přihlédnuto k faktu, že daný nástroj má být začleněn do již existující BI architektury v rámci firmy.

3 Srovnání nejdůležitějších nástrojů pro vizualizaci dat

3.1 Kritéria pro výběr nástrojů k porovnání

Vzhledem k existenci mnoha vizualizačních a analytických nástrojů pro práci s daty, bylo vzhledem k povaze a účelu práce nutné provést určitou selekci a porovnat jen ty, které vyjdou z iniciálního vyhodnocení jako perspektivní k nasazení v praxi.

Jako vhodný vstup pro základní výběr se jeví respektovaná analýza od společnosti Gartner [13], která v rámci rozdělení trhu do tzv. magických kvadrantů rozděluje jednotlivé firmy stojící za BI řešeními do čtyř kvadrantů – na lídry, vyzyvatele, vizionáře a okrajové hráče viz Obrázek 4.



Obrázek 4: Gartnerův magický kvadrant z [13].

Ve firemní praxi není zájem ani prostor zaobírat se segmentem okrajových hráčů. Ve většině společností, jejichž cílem není zkoumat nové možnosti na trhu IT, také nejsou zajímaví vizionáři ani vyzyvatelé, jež by v úvahu připadali jen ve chvíli, kdyby dané firmě nabídli unikátní funkci-

onalitu, jež by poskytla specifickou výhodu pro daný segment, v němž společnost realizuje své obchody. Obecně je ale výhodné změřením na segment lídrů.

V něm jsou jednoznačně identifikovány čtyři firmy se svými produkty: je to Microsoft se svým Power BI, Tableau s Tableau (v průběhu roku 2019 odkoupen Salesforce), Qlik s Qlik Sense a ThoughtSpot s ThoughtSpotem. Čtvrtý zmiňovaný produkt se nicméně do kvadrantu lídrů dostal poprvé a jeho hlavní síla je ve vyhledávání informací spojeném s umělou inteligencí, ne ve vizualizaci, o kterou jde v této bakalářské práci především.

V rámci České republiky a Slovenska navíc nejde o prozatím nijak vyhledávanou ani podporovanou službu. Koncem roku 2019 totiž teprve vznikla první konzultantská firma, která se zabývá prodejem a pomocí s nasazením ThoughtSpot. Jakkoliv tak může být software do budoucna zajímavou alternativou ke zvolené trojici, momentálně je kvůli nízké penetraci na trhu, nedostupnosti konzultantů a nedostatku potenciálních zaměstnanců s jeho znalostí nepříliš použitelný.

Do dalšího porovnání byly vybrány tři nástroje, jež se umístily na vrcholu žebříčku a které měly dostatečnou podporu v naší zemi. Jsou jimi Power BI, Tableau a Qlik Sense.

3.2 Porovnání v oblasti připojení k datovým zdrojům

Jednou ze zásadních charakteristik pro uvažování o výběru nástroje je možnost připojit jej k datovým zdrojům standardně využívaných ve společnosti. Společnost, s níž bylo při psaní této práce spolupracováno, primárně vyžaduje napojení k SAP BW a Snowflake, samozřejmě by mělo být připojení k různým SQL databázím a souborům typu .xls, .xlsx a .csv. Výsledek je v Tabulce 1.

Je zřejmé, že požadavek splňují všechny tři systémy, u Qlik Sense je ale nutné za konektor do SAP BW připlatit jako za rozšiřující komponentu. Vzhledem k podmínkám zkušební verze nebylo možné tuto konektivitu vyzkoušet, nicméně dle ohlasů je extraktor Qlik Sense pro SAP BW velmi kvalitní a poskytuje nejlepší uživatelský komfort ze všech tří nástrojů. Praktická zkušenost ovšem v tomto případě chybí. Pro zachování objektivity srovnání bylo přímé porovnání konektoru do SAP BW vynecháno.

Tabulka 1: Dostupnost datových konektorů.

Software/Datový zdroj	Snowflake	SAP BW	SQL databáze	Excel	CSV
Power BI	Ano	Ano	Ano	Ano	Ano
Tableau	Ano	Ano	Ano	Ano	Ano
Qlik Sense	Ano	Ano, po další platbě	Ano	Ano	Ano

V rámci dalšího testování bylo zjištěno, že Power BI i Tableau mají zvláště při čtení rozsáhlých kostek výkonové problémy a při načítání je tak nutné některé infokostky v rámci SAP BW rozdělit po charakteristikách a načítat postupně. Toto řešení je velmi komplikované a nekomfortní, reálná použitelnost je tedy u všech nástrojů nízká. Pro menší objemy dat nicméně tento

způsob využitelný minimálně v Power BI a Tableau je, Qlik Sense nebyl testován. Kdybychom dále požadavek zobecnili na požadavek podpory co konektorů do co nejvíce možných datových zdrojů, tak by vítězem bylo Power BI, druhé by bylo Tableau, třetí Qlik Sense.

V praxi nicméně v komponentě pro datovou analytiku a vizualizaci není vhodné řešit ETL, proto řešíme jen připojení k datovému skladu či tržišti. Jako nejlepší možnost pro použití datového skladu s moderním vizualizačním nástrojem se jeví přenesení reportovací vrstvy mimo SAP BW. V tomto praktickém případě jde o Snowflake databázi obsluhovanou přes cloudovou ETL a DWH platformu Keboola. Keboola v kombinaci se Snowflake totiž umožňuje připojení k mnoha primárním zdrojům dat a samotné modelování v Keboole, potažmo Snowflake, je jednoduché a probíhá pomocí SQL. Modelování v rámci SAP BW je možné jen přes několik rozhraní, trvá dlouhou dobu a přenos datového modelu do moderních vizualizačních nástrojů je komplikovaný.

3.3 Porovnání v oblasti rychlosti načtení datového zdroje

Pro tuto část porovnání byl vypracován modelový příklad načtení deseti miliónu záznamů o 20 sloupcích do datového modelu z daného softwaru a z databáze Snowflake/souboru CSV. Výsledky viz Tabulka 2.

Tabulka 2: Rychlost načtení dat z různých zdrojů.

Software/Datový zdroj	Snowflake	CSV
Power BI	3:45 min	3:20 min
Tableau	2:05 min	0:45 min
Qlik Sense	5:06 min	1:00 min

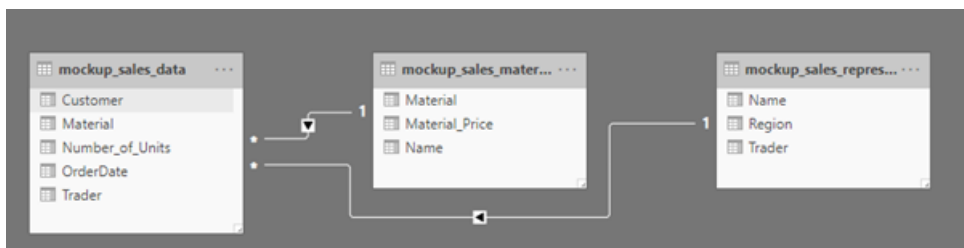
Tableau vyniká v rychlosti čtení z obou porovnávaných zdrojů dat, Power BI je téměř dvojnásobně pomalejší při čtení ze Snowflake databáze a více jak třikrát tak pomalé při čtení rozsáhlého CSV souboru. QlikSense zaostává dva a půlkrát v případě databáze Snowflake, naopak je jen o třetinu pomalejší než Tableau při čtení z CSV souboru.

V praxi je nicméně mnohem využívanější načítání dat z databází typu Snowflake. Toto je třeba zohlednit i v závěrečném hodnocení - v této kategorii tak vítězí Tableau, druhé je Power BI a třetí Qlik Sense.

3.4 Porovnání v oblasti datového modelování

Vizuálně a z programátorského pohledu klasicky se datový model utváří v Power BI. Jak je vidět na Obrázku 5, jde vlastně o klasické spojování tabulek v rámci relačního modelu. Modely následně fungují rychle a spolehlivě.

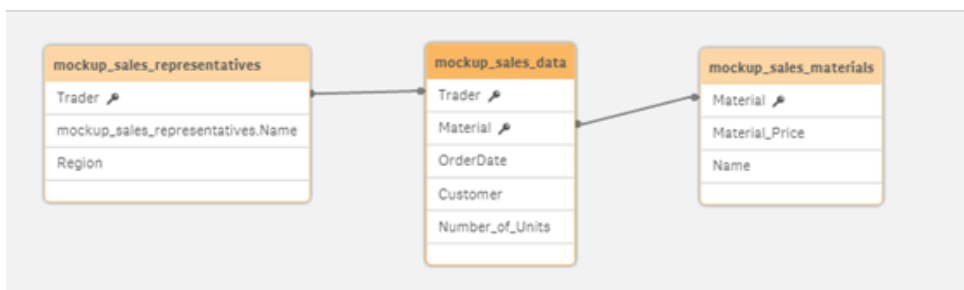
Klasického relačního modelu se drží také Qlik Sense, v rámci uživatelského rozhraní jsou ale tabulky zobrazené jako bubliny, viz 6. Pro zkušeného uživatele je přehlednější zobrazení skrze tabulky jako na Obrázku 7, jež je ale jako zobrazení dostupné až po načtení dat, nikoliv v rámci tvoření modelu.



Obrázek 5: Ukázka datového modelu v Power BI.



Obrázek 6: Bublinový datový model v Qlik Sense.



Obrázek 7: Klasický tabulkový datový model v Qlik Sense.

Subjektivně vizuálně nepřehledné je modelování v rámci Tableau, viz Obrázek 8, kde se schematicky spojují jednotlivé datové zdroje, bez zvýraznění tabulek. V rámci Tableau není možné zobrazit v rámci modelování detailní pohled na tabulky a vypsát si alespoň jejich sloupce. Při větším počtu tabulek ztrácí i pokročilý uživatel orientaci v tom, jak se jednotlivé tabulky spojují.

Nevýhodou Power BI je možnost určit pro propojení tabulek jako jednoznačný klíč pouze jeden sloupec tabulky. To často vede k umělému vytváření unikátního klíče skrze textové spojování dvou a více sloupců. To není pohodlné a uměle se tak v rámci modelu navyšuje počet



	Sort fields	Data source order				
mockup_sales_data.csv	Abc	#	mockup_sales_data.csv	#	Abc	#
OrderDate	Trader (mockup_s...	Customer	Material (mockup...	Number_of_Units	Material	Material_Price
05.01.2019	TR0001	500 001	P000001	62	P000001	

Obrázek 8: Ukázka datového modelu v Tableau.

sloupců, zvláště pokud je nutné upravit napojení na více tabulek. V porovnání s Power BI je v rámci modelování možno v Qlik Sense propojit tabulky na základě více sloupců tabulky a použít tak složený klíč, stejně tak je to možné v Tableau.

Tableau samotné ještě disponuje unikátní funkcí Blend, pomocí níž lze jednorázově v rámci vizualizace připojit tabulku k té, na níž je vizualizace postavena, bez nutnosti je spojit v modelu. Pro Blend je nutné určit jednu společnou dimenzi, což ale umožňuje vizualizaci i obtížně propojitelných dat různé granuality. Např. k detailním datům o prodejech se dá jednoduše připojit agregovaný prodejní plán. Jde o velmi příjemnou funkci pro zkušenější analytiku, její používání je ale obtížné v rozsáhlejších dashboardech, jelikož není systémové.

Celkové porovnání v Tabulce 3 vychází z hlediska přehlednosti datového modelování ve prospěch Power BI, na druhém místě je Qlik Sense, až na třetím pak Tableau. V rámci funkčnosti je na prvním místě Tableau, na druhém místě Qlik Sense a na třetím Power BI.

Tabulka 3: Datové modelování

Software	Přehlednost	Snadnost spojení tabulek	Další funkce
Power BI	Velmi přehledné	Jen přes jeden sloupec	-
Tableau	Velmi nepřehledné	Přes více sloupců	-
Qlik Sense	Přehledné	Přes více sloupců	Jednorázové připojení přes Blend

3.5 Porovnání práce s datovými typy

Práce s datovými typy znamená kategorizaci a transformaci dat, pokud je na zdroji v nevyhovujících formátech. Podporu tohoto základního zpracování nabízí všechny tři nástroje, každý z nich ale v jiné míře a kvalitě.

V rámci Power BI je transformační aparát nastaven jednoduše a nabízí několik flexibilních řešení. V rámci kategorizace nabízí široký výčet typů dat a při procesu transformace umožňuje

hromadně nahrazovat hodnoty za jiné a odfiltrovat řádky s chybnými hodnotami. Při automatickém rozpoznávání datových typů z různých zdrojů odvádí dobrou práci, jedinou nevýhodou je to, že při čtení z textových souborů se při detekci desetinné tečky a čárky řídí regionálním nastavením – lze nastavit specificky pro každý dashboard, nebo globálně. Bohužel nelze nastavit zvlášť pro každý datový zdroj, takže při existenci více různých textových zdrojů je nutné problém ošetřit např. přes nahrazení desetinné tečky čárkou.

Tableau v rámci transformačního aparátu nabízí méně než Power BI, ale v rámci nastavení a automatické detekce datových typů funguje prakticky totožně. Obtížně se v něm ale dále nahrazují chybné hodnoty, protože v tomto případě je nutné využít rozšiřující datový nástroj Tableau Prep. Jeho využití ale znamená mnoho kroků navíc a ovládání je pro běžného uživatele obtížné. Výhodou Tableau oproti Power BI je možnost nastavit znakové sady, oddělovače a další pro každý datový zdroj zvlášť.

Qlik Sense v rámci kategorizace a transformace dat nabízí jen čtyři typy: obecný, datum, časové razítko a geografická data. Pod obecný typ jsou zařazena jak čísla, tak texty a další. Může být obtížné zachovat specifické prvky z datových skladů (například číselné ID s úvodními nulami), ale vše se dá v rámci zpracování datového zdroje nastavit. Pro Qlik Sense typicky je nicméně nutné se více ponořit do specifického jazyka a pomoci si programováním skriptů. Při automatickém rozpoznávání a loadu dat umožňuje Qlik Sense stejně jako Tableau měnit znakové sady, oddělovače a další nastavení pro každý datový zdroj zvlášť.

V rámci porovnání tak i přes drobnou nevýhodu v automatické detekci dat vítězí Power BI, následováno Qlik Sense a až na třetím místě končí Tableau, které pro pokročilé funkce vyžaduje použití Tableau Prep.

3.6 Porovnání možností rozšíření datového modelu pomocí vytváření tabulek, sloupců a metrik

Častým požadavkem je v rámci vizualizačního nástroje dopočítat nějakou metriku tak, ať nezatěžuje datový model v datovém tržišti nebo dokonce přímo skladu – jako příklad lze použít objem tržeb ve chvíli, kdy je z DWH zřejmé, kolik kusů výrobku bylo prodáno a jaká je ceníková cena. Často také vzniká požadavek na dopočtení celé tabulky pro využití ve specifické analýze či modelu, nebo její operativní vytvoření přes manuální vstup (např. při vytvoření jednoduchého číselníku).

V první řadě je třeba definovat rozdíl mezi sloupcem a metrikou: metrika má jasně danou agregaci, což následně slouží ke správnému reportingu v různých úrovních granularity dat. Dopočtený sloupec je jen sloupec v rámci tabulky.

Power BI umožňuje jednoduché tvoření nových sloupců i metrik přes využití jazyka DAX nebo M. M se podobá klasickému programovacímu jazyku, kdežto DAX je velmi podobný vzorcům v Excelu. V rámci v úvodu kapitoly uvedeného příkladu by vypadalo dopočtení metriky jako na Obrázku 9. Nový sloupec je do modelu doplněn pomocí selekce jedné hodnoty z první

tabulky a pronásobení druhou hodnotou, kterou je přes klíčové slovo RELATED dohledána v tabulce spojené jednoznačným klíčem.

V tomto případě je správnější kalkulaci provést v rámci metriky, nicméně Power BI je v tomto benevolentní a umožňuje daleko více doplňovat i samotné sloupce, což může být ve specifických případech výhodné.

```
TotalSales = mockup_sales_data[Number_of_Units]*RELATED(mockup_sales_materials[Material_Price])
```

Obrázek 9: Ukázka syntaxe vytvoření nového sloupce v Power BI.

Tvorba celých nových tabulek je v Power BI umožněná, přičemž je k dispozici jak možnost využít kódu v M, tak si pomoci automatizovanými funkcemi. Manuální vyplnění jednoduché tabulky či číselníku je pak otázkou několika kliknutí a funguje spolehlivě.

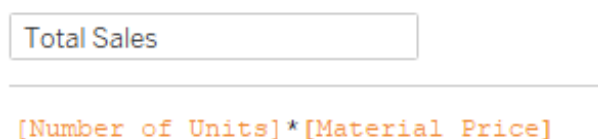
QlikSense naopak vyžaduje specifikaci pomocí metriky v tomto případě vyžaduje, takže je nutné využít stupeň agregace, abychom s daty mohli v rámci vizualizace pracovat (u Power BI je to pouze doporučené). Proto je při dopočtení pole nutné specifikovat, jak jej uživatele ve výsledku chce zobrazit – jako sumu, jako počet atd. Vzorec viz Obrázek 10

```
=SUM(Material_Price*Number_of_Units)
```

Obrázek 10: Ukázka syntaxe vytvoření nové metriky v Qlik Sense.

Co se týče tvoření celých tabulek nebo jejich tvorby z manuálního vstupu, je v Qlik Sense možno tabulku vytvořit pomocí kódu bez výrazné nápovědy, takže je tato tvorba složitá. Manuální vstup je ale stejně pohodlný a jednoduchý jako u Power BI.

Tableau je v rámci tvoření nových sloupců a metrik nejbenevolentnější: stačí vybrat dvě pole, které chcete pronásobit, a to je vše, viz Obrázek 11. Spojování přes Related zde nenajdete, vše si engine řeší sám, což může být výhoda i obrovská nevýhoda, když chcete specifickou funkčnost. Se složitějšími modely ale dokáže být jednoduchost poněkud dvojsečná (chcete například nastavit specifickou agregaci a nemůžete) a opět přichází na řadu nutnost použít Tableau Prep nebo zasáhnout do datového modelu.



```
Total Sales
```

```
[Number_of_Units] * [Material_Price]
```

Obrázek 11: Ukázka syntaxe vytvoření nové metriky v Tableau.

Tvorba celých nových tabulek je v Tableau možná opět pouze v rozšíření Tableau Prep, stejně jako manuální vstup do datového modelu.

Celkové shrnutí je k dispozici v Tabulce 4 a jako nejlepší nástroj se jeví Power BI, na druhém místě Qlik Sense, na třetím Tableau.

Tabulka 4: Vlastnosti rozšiřování datového modelu v jednotlivých nástrojích.

Software	Power BI	Tableau	Qlik Sense
Jednoduchost tvorby	Jednoduché	Velmi jednoduché	Středně složité
Vynucování best-practice	Nízké	Velmi nízké	Vysoké
Možnosti customizace	Velmi vysoké	Nízké	Střední
Vytváření tabulek	Možné, vizuální i kód	Jen v Tableau Prep	Možné, jen kód
Manuální vstup v DM	Možné, jednoduché	Jen v Tableau Prep	Možné, jednoduché

3.7 Porovnání dostupnosti vývojového prostředí

Dostupnost vývojového prostředí na jednotlivých operačních systémech je shrnuta v Tabulce 5, přičemž prohlížení hotových reportů je u všech nástrojů možné přes libovolný internetový prohlížeč.

Tabulka 5: Možnost vývoje na různých operačních systémech.

Software	Power BI	Tableau	Qlik Sense
Windows	Ano	Ano	Ano (cloud přes prohlížeč)
Mac OS	Ne	Ano	Ano (cloud přes prohlížeč)
Linux	Ne	Ne	Ano (cloud přes prohlížeč)

Z hlediska dostupnosti vývojového prostředí je nejuniverzálnějším nástrojem Qlik Sense, neboť veškerý vývoj je u něj směřován na cloud. Je tedy dostupný z jakékoliv platformy a komfort vývoje ovlivňuje rychlost připojení, a ne HW vybavení klienta. Při testování se nicméně stávalo, že se po chvíli nečinnosti vše odpojilo a přes nicneříkající chybové hlášky bylo nutno se odhlásit a zase přihlásit. Jednou se toto stalo i při práci (vytváření modelového dashboardu) a mělo za následek ztrátu části práce.

Pro vývoj v Tableau je nutné používat Tableau Desktop, jež je narozdíl od Qlik Sense ovlivněn výkonem PC. Tableau je nicméně dostupné jak na systému Windows, tak MacOS a umožňuje vývoj většímu množství lidí.

Pro vývoj na platformě Power BI je nutné používat Power BI Desktop, jehož výkon je ovlivněn výkonem PC. Power BI Desktop je k dispozici pouze na systému Windows. Dostupnost má tak Power BI nejhorší.

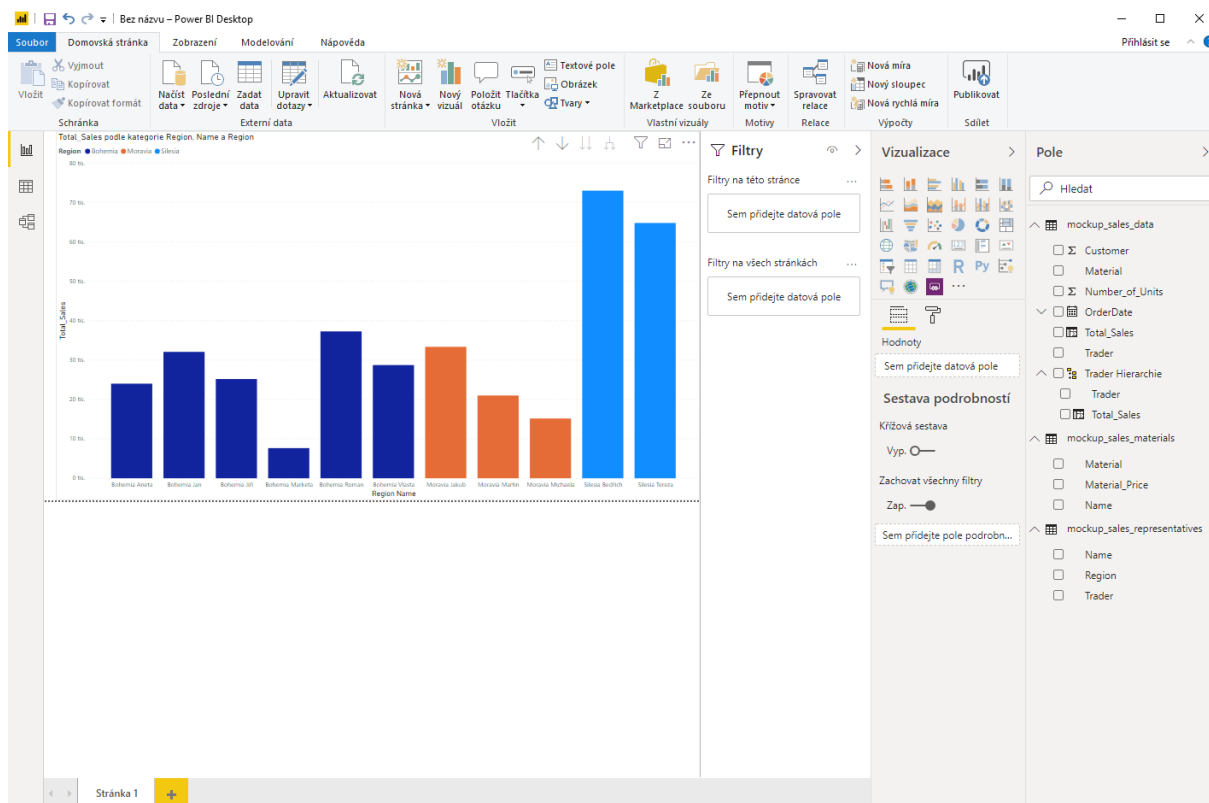
BI oddělení a IT většinou pracují právě na Windows platformě, jen servery běží na Linuxu, a MacOS se kvůli nákladům vyskytuje jen u výše postavených manažerů společností, u nichž není předpoklad vytváření reportů, ale pouze jejich sledování. Kategorie je tak bez hodnocení.

3.8 Porovnání ovládání přístupu k datům

Základním požadavkem ve společnostech pro uplatnění autorizací a řízení přístupu k reportům je možnost tyto role vztáhnout k používanému softwaru. Nejčastěji jde o Microsoft Active Directory, které je v rámci korporátů standardem k řízení přístupů. Jak Qlik Sense tak Power BI tyto funkcionality nabízejí, bohužel Tableau zcela chybí a role je nutné administrovat jen v Tableau konzoli.

3.9 Porovnání tvorby vizualizací

Power BI umožňuje vytváření vizualizací na stránce dashboardu. Systém vytváření je pak jednoduchý a uživatelsky přívětivý: do jednotlivých kategorií hodnot unikátních pro každý typ grafu je nutné umístit charakteristiku, či metriku. Detailnější nastavení je pak možné v dalších podmenu, kde lze měnit barvy, popisky os a další, podobně, jako je tomu například v Excelu, jehož rozhraní Power BI vizuálně připomíná, viz Obrázek 12.



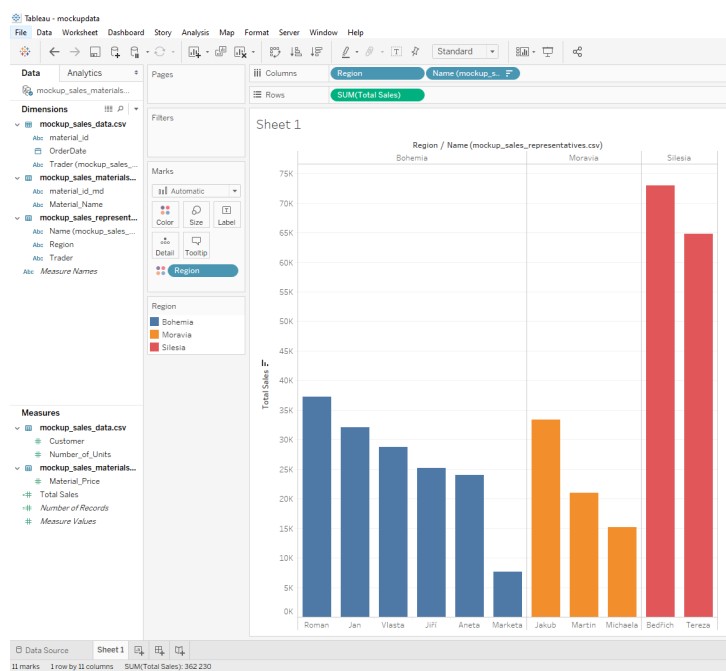
Obrázek 12: Ukázka jednoduché vizualizace v Power BI.

V rámci Power BI je dále jednoduše možné měnit typ grafu a rozšířit možnosti týkající se zobrazení dat i formou stažení doplňkových grafů, nebo vývojem vlastního doplňku, jež unikátní graf do nabídky přidá. Jednou z nevýhod, které Power BI provázejí, je nemožnost seřadit sloupce

podle více hodnot za sebou (v tomto případě region a následně hodnota obrátu) – Tableau i Qlik Sense tuto možnost mají.

V Power BI se dá toto omezení obejít v některých případech skrze vytvoření sloupce Index přímo v datovém modelu, nicméně to vyžaduje kvůli pouhému řazení měnit datový model, což nelze doporučit. V poslední aktualizaci z dubna 2020 se vícenásobné řazení nicméně objevilo jako nová funkcionality pro vizualizace typu tabulka, což by mohlo naznačovat, že se blíží i do jiných vizualizací.

Tableau oproti tomu nabízí tvoření vizualizací samostatně, ne jako součást dashboardu. V rámci tvorby se také pomocí unikátního enginu snaží našeptávat, jaká vizualizace je pro daný soubor hodnot nejvýhodnější.

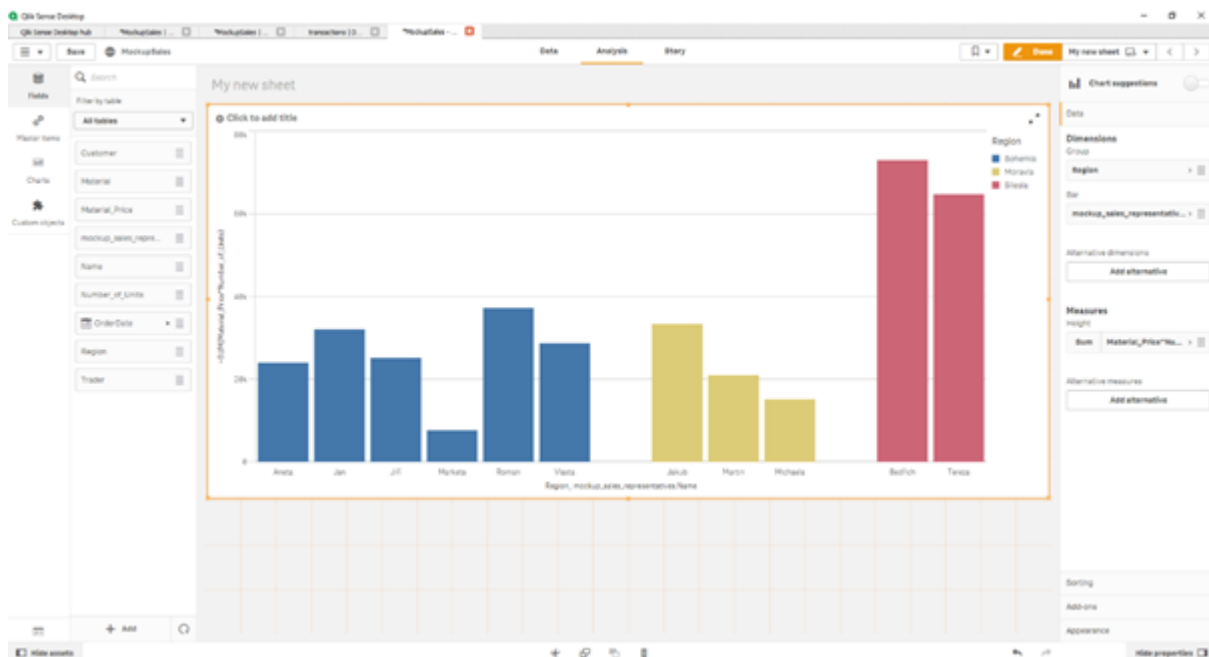


Obrázek 13: Ukázka jednoduché vizualizace v Tableau.

Základní činností při tvorbě vizualizací je vyplňování charakteristik do sloupců a řádků, přičemž každou lze dále upravit skrze detailní specifikaci, jestli brát hodnotu jako diskrétní nebo spojitou, jakou část vzít z datumu apod. Základní rozložení uživatelského rozhraní je znázorněno v Obrázku 13. Vzhledem k jednoduchosti a rychlosti celého procesu, je Tableau nejpříjemnější pro vytvoření jednoduché vizualizace. Na rozdíl od Power BI pak nejsou jednotlivé možnosti specifické pro grafy, ale např. možnost dělení barev je dostupná u všech výchozích grafů, což je obrovské plus.

Tableau stejně jako Power BI umožňuje rozšířit možnost izobrazení dat skrze stažení doplňkových grafů a vývoj vlastních doplňků. Nabídka již hotových vizualizací vytvořených komunitou je ale o něco menší.

V rámci Qlik Sense je podobně jako v Power BI spojeno vytváření vizualizací s organizací vizualizace na dashboardu. Na rozdíl od Power BI a podobně jako v Tableau ale algoritmus na základě dat našeptává, jaká vizualizace se pro daný soubor dimenzí a metrik nejlépe hodí. Qlik Sense se dále místo toho, aby po uživateli vyžadovalo umístění do speciálních kategorií nebo aby vyžadovalo jednoduché určení, co má být ve sloupcích a v řádcích, vyžaduje označení metrik (číselné hodnoty) a dimenzí (kategorie). Vizualizace pak v praxi vypadá jako na Obrázku 14.



Obrázek 14: Ukázka jednoduché vizualizace v Qlik Sense.

Qlik Sense je dále specifické tím, že pro změnu barev je třeba vytvořit tzv. „Master“ vizualizaci a tu samostatně ukládat. Stejně tak pro každou drobnost je nutné navigovat do speciálního zanořeného menu. V něm sice je k dispozici detailní nastavení, ale i jednoduché věci vyžadují precizní nastavení mnoha parametrů, které mohou být pro běžného uživatele matoucí.

Qlik Sense, stejně jako Power BI a Tableau, nabízí krom široké nabídky základních vizualizací možnost stáhnout některá komunitní rozšíření, řada těch lepších je nicméně zpoplatněná a není jich tak velký počet jako v případě Power BI a Tableau.

Z hlediska uživatelské přívětivosti i možností týkající se jednotlivých vizualizací je vítězem v tomto případě Tableau, následované Power BI a Qlik Sense.

3.10 Porovnání procesu tvorby dashboardů

Vytváření dashboardů je z uživatelského hlediska nejpříjemnější v Power BI, těsně následované Qlik Sense. V Power BI jsou jednotlivé vizualizace umísťovány přímo na dashboardový sešit a od začátku je vytvářen dashboard – přestože může být celý dashboard tvořen jedinou vizuali-

zací. Designování je intuitivní a skvěle funguje například křížové filtrování, které je od základu zapnuté, a při využití jednoho filtru u jedné vizualizace jej aplikuje i na další grafy.

Proces tvorby je v Qlik Sense podobný: na plochu dashboardu jsou umisťovány jednotlivé vizualizace. V prvních chvílích se uživatel snaží program automaticky pomáhat a vizualizace vzájemně třídit, při vývoji zkušebních dashboardů se ale osvědčilo pomocníky vypnout.

Co se týče samotného menu, působí o něco složitějším dojmem, jako by se snažilo nabídnout co nejvíce funkcí v co nejvíce tabulkách. Výsledek je ale přinejlepším matoucí. Dost nepohodlný je například fakt, že i pro změnu barvy je nutno vytvořit „master“ vizualizaci, kterou lze až po samostatném uložení umístit do dashboardu jako pevný objekt.

A pokud je například třeba upravit barvu grafů do korporátních, je nutné opět předělat všechny jednotlivé master vizualizace místo jednoduché změny v nastavení zobrazení. Filtrování napříč vizualizacemi ale funguje velmi dobře.

V rámci Tableau je vytváření dashboardů řešeno tak, že do dashboardu se postupně skládají jednotlivé již vytvořené vizualizace, v rámci terminologie Tableau Worksheets. Ty lze systémem „přetáhni a pusť“ umístit na dashboard, kde se dle automatického rozpoznávání Tableau skládají do sloupců i řádků a následně tvoří pevně dané bloky. Vytvořit jednoduchý dashboard je v Tableau složité a frustrující, neboť pomoc systému v tomto případě vše spíše zhoršuje. V rámci dashboardu je nutno navíc nastavit křížové filtrování pro každý objekt zvlášť – to znamená, že se musí přesně stanovit, že zafiltrování v grafu A ovlivní hodnoty pro grafy B a C.

Na prvním místě v této kategorii se umisťuje Power BI, na druhém místě Qlik Sense, na třetím Tableau.

3.11 Porovnání nákladů na uživatele

Ceny za licenci se u Power BI, Tableau i Qlik Sense liší a každá firma má trochu jiné podmínky k jejich využití, nicméně u všech se cena váže na měsíc používání a na uživatele.

Power BI nečlení uživatele do žádných skupin a cena za licenci a měsíc je pevně daná - licence jde ale pružně z měsíce na měsíc aktivovat a deaktivovat. Tableau oproti tomu člení uživatele do tří kategorií: administrátor, pokročilý uživatel využívající platformu pro vlastní analýzy a konzument reportů. Qlik Sense uživatele dělí jen na administrátory a pokročilé uživatele. Jednotlivé ceny za osobu a měsíc jsou znázorněny v Tabulce 6

Tabulka 6: Ceny za licenci a měsíc dle [14], [15] a [16]

Software	Power BI	Tableau	Qlik Sense
Administrátor	10 USD	70 USD	70 USD
Pokročilý uživatel	10 USD	35 USD	40 USD
Konzument reportů	10 USD	12 USD	40 USD

V spolupracující společnosti, kde byla tato práce vypracována, byla představa o architektuře následující: 5 lidí, co budou administrátoři a budou vytvářet datové modely a reporty.

Dalších cca 20 lidí, co budou tvořit ad-hoc reporty a využívat administrátory vytvořené datové zdroje. Následně dalších cca 200 lidí – konzumentů, kteří budou využívat již vytvořené reporty a interagovat s nimi pouze uživatelsky.

Jak je vidět, agresivní cenovou politikou jednoznačně vítězí Power BI. Pokud společnost navíc využívá balík Microsoft Office, je cena za Power BI zahrnuta již v něm. Tableau se jako jediné snaží odstupňovat jednotlivé uživatele, ale cena je i u nejnižší licence vyšší než u Microsoftu. QlikSense nasazuje dvě sazby – 70 a 40 USD za osobu [16].

Vedle těchto licencí jsou k dispozici v případě Microsoftu ještě tzv. Premium licence, kdy se platí měsíční paušál 4995 USD [14] za neomezený počet uživatelů a je omezeno jen místo na disku, plus jsou odemčeny ještě další funkce. Qlik Sense rovněž tomu nabízí pouze Business variantu, vhodnou pro nasazení napříč týmy v organizaci, kde požaduje paušál 30 EUR za osobu [16].

Cenově tak jednoznačně vítězí Power BI, druhé je Tableau a třetí je Qlik Sense.

3.12 Další výhody a nevýhody jednotlivých nástrojů v bodech

3.12.1 Power BI

- Nejlepší přehlednost zanořování v rámci hierarchií – můžete se podívat na větší detail se zachováním rozdělení na vyšší úrovni i s potlačením.
- Export dat jen do PDF, export podkladu nefunguje dostatečně spolehlivě.
- Velká výhoda – možnost napojení na datový model z Excelu a vytváření kontingenčních tabulek. Skvělé jako jeden zdroj pro analytiky.
- Nevýhoda ve formě omezeného množství vizualizačních bodů (patrné zvláště na mapových a Gantt vizualizacích) – Power BI upozorňuje jen nenápadnou ikonou, že nezobrazuje všechna data.

3.12.2 Tableau

- Možnosti dalšího pohledu dle hierarchie podle toho, jak jej nastaví tvůrce reportu.
- Při vytváření statistického binu (např. věková skupina 0-15, 16-30, 31-45...) je možno nastavit, že velikost binu půjde z dashboardu změnit.
- Export grafu i podkladových dat na jedno kliknutí.

3.12.3 Qlik Sense

- Filtrování jako hledání na google, přehled toho, co je zafiltrováno, v horní části obrazovky.
- Celkový dojem z ovládání je takový, že pokud je třeba cokoliv, je třeba to naskriptovat.

3.13 Vyhovující datová základna pro moderní analytické nástroje

Jedním z nejdůležitějších zjištění při porovnávání nástrojů Power BI, Qlik Sense a Tableau, je, že moderním vizualizačním a analytickým více vyhovuje pracovat s minimálně částečně normalizovaným datovým zdrojem než s klasickými velmi robustními denormalizovanými reportingovými kostkami. Při praktickém testování byl nad částečně normalizovaným modelem rychlejší jak vývoj, tak následný reporting – kde se odezva při shodném počtu záznamů v prvním případě pohybovala v řádech sekund, v druhém případě byla podstatně rychlejší. Jako ideální se pak jevílo v teorii popsané hvězdicové schéma 2.2.4.

Pro přehlednost a rychlost nicméně z praktické zkušenosti vyšlo najevo i to, že se vyplatí nebyť při normalizaci příliš striktní. Oddělit hlavní tabulky s transakčními daty rozhodně má smysl, ale několikanásobně vrstvit kmenová data kvůli vzájemné závislosti atributů je již spíše kontraproduktivní – a to u všech nástrojů.

3.14 Shrnutí

Tableau je výborný software, který vyniká v samotných jednotlivých vizualizacích, intuitivním UI a výborně se s ním pracuje v roli samostatného analytika. Pokud by výběr zněl „software pro konzultanta nebo analytika připravujícího prezentace“, byl by Tableau na špici. Sráží ho však složitá výstavba dashboardů, nepřiliš dobré ošetření budování datového modelu (pokud se ale udělá jinde, funguje výborně), nevyřešený přístup k datům a také nejasná budoucnost: Tableau bylo koncem roku 2019 odkoupeno společností Salesforce a další vývoj softwaru není jasný. Pro korporaci je tak bohužel v současné době nedoporučitelný.

Qlik Sense oproti Tableau nabízí hodně zajímavých funkcionalit i pro korporátní sféru, nicméně hodně ztrácí jak v ceně, tak v uživatelské nepřívětivosti. Jde v něm udělat skoro všechno, ale pokaždé je to skrze nespočet menu, a i jednoduchá změna barev je složitá. Nahrání datového modelu, vytváření metrik... to vše je oproti ostatním nástrojům složitější a připomíná programování. K plnému využití výhod, jež tento přístup nabízí, je třeba několik let zkušeností. Velkou nevýhodou je také extrémně vysoká cena.

Power BI je v některých ohledech velmi silné (dopočítávání metrik, možnost napojení do Excelu, jednoduchá tvorba dashboardů, známé prostředí, dobrá dokumentace a rozsáhlá komunita), v některých až nepochopitelně slabé (nemožnost multi-sloupcového seřazování hodnot, napojení skrze jeden primární klíč). Jako celek se ale jeví jako nejvíce použitelné vzhledem k velmi nízké ceně a celkově velmi dobré funkcionalitě. Společnost proto v rámci své strategie zvolila na základě této analýzy a výběrového řízení právě Power BI jako nový vizualizační nástroj.

4 Případová studie – vytížení vysokozdvížných vozíků

K demonstraci využití a potvrzení správnosti výběru vizualizačního a analytického nástroje Power BI byla zvolena oblast spojená s nájedem a využitím vysokozdvížných vozíků v konkrétní výrobní firmě. Poskytnutá data jsou stoprocentně reálná, jedinou úpravou je anonymizace údajů o řidičích, kdy konkrétní jména byla nahrazena vygenerovaným ID.

Případová studie je logicky členěna na popis počáteční situace, extrakci dat, jejich transformaci, nahrání do vizualizačního nástroje, jejich vytížení a vizualizaci.

4.1 Počáteční situace

V rámci řízení skladů vznikl požadavek na monitoring využití jednotlivých vysokozdvížných vozíků i jejich řidičů. Vysokozdvížné vozíky od firmy Toyota, využívané ve firmě, poskytují jak monitorovací jednotky pro sběr požadovaných dat, tak i vlastní systém pro management flotily nazývaný ISite.

Celkový součet
Provozy: 6 Stroje celkem: 114 Využití stroje: 23,13%
Nárazy stroje: 15,45 Řidiči celkem: 474 Využití řidičů: 8,27%
Nárazy řidičů: 3,72

Název provozu ▼ 1 2 3 4 5 6

Název provozu	Stroj			Řidič		
	Počet	Využití	Průměr nárazů	Počet	Využití	Průměr nárazů
Závod 1	13	31,48%	39,15	87	4,70%	5,85
Závod 2	25	34,06%	17,76	116	8,71%	3,83
Závod 3	20	23,35%	15,30	53	9,10%	5,77
Závod 4	28	16,93%	11,57	71	14,35%	4,56
Závod 5	17	15,22%	2,12	90	5,41%	0,40
Závod 6	11	25,13%	12,91	57	7,47%	2,49

Obrázek 15: Ukázka jednoduchého reportu s anonymizovanými názvy provozu v systému ISite.

Ten, dle firemních webových stránek, disponuje řadou funkcí včetně informací o využití vozíků, řidičů, reporty a tabulkami [17], nicméně jeho provedení je v současné době spartanské. Pro správu vozíků a např. nastavení blokace při určité intenzitě nárazu, slouží dobře, ale reporting samotný je pouze textový, tabulkový a pro větší detail je nutné jít spíše cestou stažení surových dat do Excelu a reportingu v něm. Ukázky reportů s anonymizovanými údaji lze vidět na Obrázku 15.

Reporting byl vzhledem ke své neefektivnosti pro společnost nedostatečný, vytěžení dat zabíralo pověřenému pracovníkovi mnohdy celý jeden pracovní den v týdnu. Cílové zpracování bylo to jako jeden z prvních ukázkových reportů v moderním vizualizačním nástroji, vybraném na základě porovnání provedeném v kapitole 3 této práce.

4.2 Cílový koncept

Cílem je vytěžit data ze systému ISite a vizualizovat je tak, ať vyhovují požadavkům rychlého vizuálního reportingu pro manažery a vedoucí jednotlivých skladů. V rámci nájezdu vozíků je žádoucí zobrazovat průměrný nájezd vozíků v jednotlivých kategoriích a závodech, nějakým způsobem vizualizovat to, jak vozíky jezdí zároveň (s cílem zjistit, zda se nedá efektivnějším využitím vozíků snížit jejich počet), umožnit detailní prozkoumání nájezdu jednotlivých vozíků a časů s nimi spojených. Také zobrazit jejich vytíženost vzhledem ke kontraktu.

Obdobný požadavek byl vznesen také k vytíženosti řidičů a porovnání času směny s časem jízd. Vizualizace by také měla ukázat ve vztahu k řidičům, jak moc s vozíky bourají, a jak časté to je ve vztahu k naježděným hodinám.

Veškerý reporting by také dle požadavku měl být veden rámci toho, že noční směna začínající např. v pondělí 22:00 bude reportována do pondělního dne celá, tedy do úterý 6:00.

Cílem procesu pak je naprogramovat většinu důležitých transformací v rámci ETL nástroje, a v rámci vybraného vizualizačního nástroje (Power BI) primárně vizualizovat, případně dopočítávat jen velmi jednoduché údaje.

4.3 ETL proces a nahrání do cílové databáze

Obecně nejlepší formou získávání dat z klasických databází a jejich importu do reportovacích databází, je využití ETL nástroje či skriptu pro upload do cílové destinace, a na zdrojové straně vytvoření speciálního databázového view přímo na základu specifikace klienta, či zabezpečený přístup skrze API. K jeho využití je nicméně nutné, aby zdrojový systém tato připojení umožňoval.

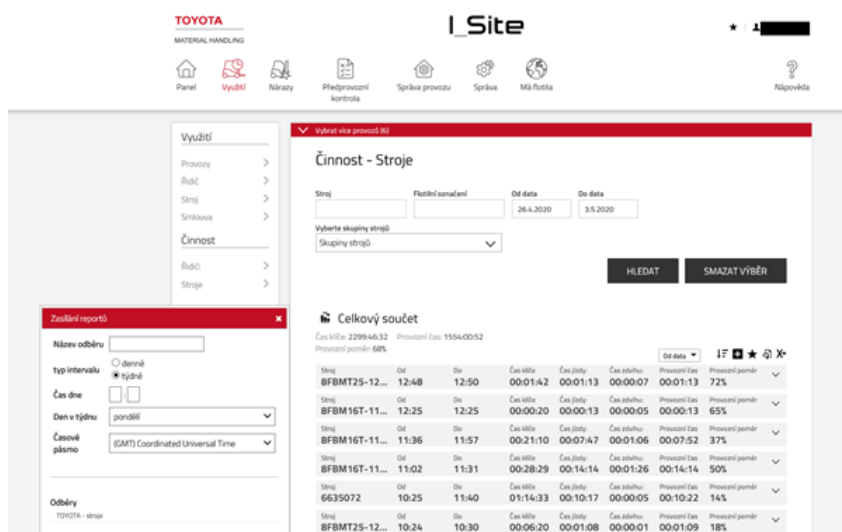
4.3.1 Zdroje dat a napojení na ně

Díky interview s manažery i klíčovými uživateli reportingu byly identifikovány dva zdroje dat: pro všechna transakční data a část master dat šlo o interní systém firmy Toyota zvaný ISite, do jehož databáze jsou v pravidelných a nastavitelných intervalech v rozmezí řádů hodin až dní importovány údaje ze snímacích jednotek umístěných na jednotlivých VZV.

Malou část master dat chtěli uživatelé ještě přidat, resp. přidat další vlastní členění vysokozdvížných vozíků. Jelikož nešlo rozšířit informace přímo ve zdrojovém systému, bylo navrženo jejich vyplnění do Tabulek Google. Tabulka Google byla navržena proto, že pro účely jednoduché údržby bez zásahu IT se jeví jako nejjednodušší způsob, mimo naprogramování speciální aplikace. Tabulka Google je pak vhodná i proto, že umožňuje stažení dat skrze API.

Systém Toyota ISite pak bohužel stažení dat skrze API neumožňuje, stejně tak jako se dodavatel postavil zamítavě k vytvoření view. ETL tak muselo být navrženo jinak: v systému je možnost zasílat .XLS reporty dle definice předpřipravené dodavatelem. V případě ISite nicméně „report“ znamená v případě exportu reportu v největším detailu prostý export podkladových dat a jeho zaslání na email. Toto nepříliš standardní řešení nicméně umožňuje strojově automatizovanou extrakci dat a vyhovuje vstupnímu požadavku.

V tomto konkrétním případě tak proběhlo nastavení zasílání reportů o detailech jednotlivých jízd VZV, nárazech s VZV v denní periodicitě a měsíční zasílání údajů o kontraktech k jednotlivým vozíkům – nastavení je uživatelsky přívětivé, viz Obrázek 16.



Obrázek 16: Ukázka nastavení zasílání reportu v systému ISite.

Zasílání bylo konkrétně nastaveno pro reporty Činnost – Stroje, Detaily nárazů a Využití – Smlouva jejichž ukázky jsou součástí přílohy.

4.3.2 Obecné způsoby extrakce dat

Přetransformovat data z .XLS souboru do cílové podoby v databázi lze několika cestami, z nichž nejtypičtější je použití a CRON orchestrace jednotlivých extrakčních skriptů v libovolném jazyce, či sofistikovanější využití specializovaného ETL nástroje nebo platformy.

Při ověřování možné funkčnosti řešení byl vyvinut Python skript, který přes API komunikoval s Gmail klientem, vybral ze schránky emaily odeslané z Toyoty a stáhl přílohový .XLS soubor. V rámci stejného skriptu byl potom za použití veřejně dostupné knihovny csv vytvořen i .CSV soubor s uloženými daty, jež se následně skrze interface ze serveru importoval do produkční databáze Snowflake. Skript posloužil jako „proof-of-concept“, ale na produkční řešení nesplňoval bezpečnostní, a především údržbové standardy.

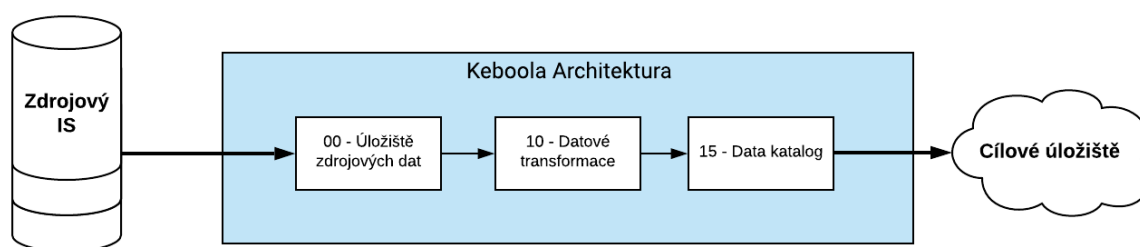
Vzhledem k tomu, a hlavně také s přihlédnutím k potřebě data dále transformovat a nutnosti napojení na různé datové zdroje, bylo preferováno komerční řešení. V tomto konkrétním případě jde o využití cloudové ETL platformy Keboola.

4.3.3 Představení Keboola

ETL platforma Keboola komplexně pokrývá datové operace v rámci typické ETL. Obsluha zahrnuje jak webovou aplikaci, tak možnost (a pro některé pokročilé konfigurování i nutnost) využívat aplikaci Postman pro volání API, autorizované zabezpečovacími tokeny. Hlavními částmi platformy jsou extraktory, transformace, writery a samotné uložení všech těchto dat.

Extraktory umožňují připojení ke zdrojům dat – typicky jde o databáze softwaru třetích stran, exportované .CSV z interních systémů a vlastních aplikací a další. Transformace umožňují s těmito daty manipulovat. V případě Kebooly lze využít k transformaci dat jazyk SQL na Snowflake základu, Python či R. Kombinace více návazných transformací v různých jazycích je rovněž možná a při pokročilejších úkolech přináší zajímavé možnosti.

Transformace si vyvíjí každý sám dle účelu, v případě extraktorů a writerů jsou připraveny již nachystané komponenty jak od autorů platformy, tak od dalších přispěvatelů. V případě chybějícího napojení je možné si nechat naprogramovat komponent na zakázku, nebo jej vyvinout svépomocí v Pythonu.



Obrázek 17: Multiprojektová struktura v Keboola.

Velkým plusem Kebooly je i možnost zapojit se do vývoje a komunikovat s vývojáři, jelikož všechny dodělávané moduly do Kebooly mají kód volně přístupný na platformě Github. Lze si tak ověřit detailní funkčnost, navrhnout konkrétní změny kódu a hlásit bugy, přičemž dle zkušeností autoři na tyto podněty většinou reagují velmi pružně.

V rámci korporátního využití je žádoucí využít primárně předpřipravené moduly, které používá více lidí, jsou v určitém standardu a podpora je ideálně dostupná přímo od dodavatele celé platformy. Zvláště u extraktorů stojí ještě za zmínku možnost využít v rámci jednotlivých konfigurací tzv. procesorů – skriptů, které mají za úkol po základním stažení dat udělat ještě nějakou další úpravu. Příkladem může být např. převod souboru z XLS do CSV formátu, přeskocení několika řádků .CSV souboru, nebo vytvoření hlavičky tak, aby ji byl systém schopen načíst.

Co se týče samotné architektury BI v rámci Kebooly, je na základě doporučení vývojářů ve firmě využívána struktura multiprojektová, jejíž základní schéma je k dispozici na Obrázku 17. Jednotlivé projekty mají následující využití:

- 00 – Úložiště zdrojových dat – zde jsou k dispozici všechny extraktory a úvodní transformace, v nichž se jen čistí data – převádí se na správné datové typy, text se transformuje do řádků tabulky atd. Vstupem je ideálně plný detail tabulky zdrojového systému nebo view. Výstupem jsou data s názvy polí jako ve zdrojovém systému, ale již připravená k dalšímu zpracování.
- 10 – Datové transformace – pouze transformace dat, kdy v tomto projektu dochází k výběru polí, která ze zdroje potřebujeme pro současný reporting a jsou v současné době využívána. Používá se ke všem složitějším datovým operacím.
- 15 – Datový katalog – tabulky v konečné podobě pro reporting, writery do cílových destinací – datových tržišť. Transformace již jen kosmetické (v současné době např. pro starý vizualizační nástroj ve firmě) a jen pro účely např. jiného formátu pro specifický výstup.

V rámci projektů se jednotlivé tabulky dále seskupují do tzv. bucketů, což jsou shluky tabulek vytěžené např. stejným extraktorem, nebo zpracované stejnou skupinou transformací. V případě transformací je pak možné nastavit, do jakého výstupního bucketu se tabulky uloží.

4.3.4 Využití ETL nástroje Keboola ke stažení dat

Keboola umožňuje extrakci dat tak, že v základním režimu ji zvládnou i netechničtí uživatelé. Stačí v menu projektu zvolit položku Components a v rozbalovacím menu vybrat Extractors. Odtud vede odkaz na seznam již existujících konfigurací extraktorů, nicméně na horní straně je zelené tlačítko „Add new extractor.“ Po kliknutí se objeví seznam již existujících extraktorů do různých systémů.

Jejich nastavení se různí, ale základ zůstává stále stejný, nejdříve je třeba autorizovat přístup k datům: u FTP je to např. jméno a heslo či private key pro SSH tunel, nicméně v modelovém příkladu Gmail extraktoru je to nutnost přihlásit se ke Google účtu.

V rámci konfigurace lze doladit nastavení ve formátu JSON: u Gmail extraktoru je to třeba hledání konkrétního textového řetězce. V rámci konfigurace je pak možné upravovat také sekci processors – a to buď přímo v konfiguraci dané komponenty, pokud to její autor umožní, nebo skrze volání API update configuration. Jak vypadá takový update konfigurace u extraktoru skrze volání API je v příkladu přímo od Keboola [18].

V případě Gmail extraktoru jsou procesory dostupné již z webového rozhraní a jejich definice do sekcí before (před extrakcí/průběhem aplikace) a after (po extrakci/aplikaci) je poměrně jednoduchá.

Zajímavé je, že procesory se za sebe dají řetězit a můžete tak s použitím základních stavebních bloků vyřešit i složitější zadání. Všechny dostupné procesory, extraktory, i writery pak získáte

opět skrze volání API např. v Postmanovi a to příkazem Get components jehož příklad je v tomto zdroji [19]. Příklad takového zřetězení procesorů je uveden ve Výpisu 1.

```
1 {
2   "before": [],
3   "after": [
4     {
5       "definition": {
6         "component": "jakub-bartel.processor-xls2csv"
7       }
8     },
9     {
10      "definition": {
11        "component": "keboola.processor-skip-lines"
12      },
13      "parameters": {
14        "lines": 5,
15        "direction_from": "top"
16      }
17    }
18  ]
19 }
```

Výpis 1: Řetězení procesorů v extraktoru Keboola

Po úspěšné extrakci dat následuje jejich uložení do tabulky v již zmíněném bucketu, který je u extraktorů identifikován jednoznačným ID konfigurace.

4.3.5 Transformace dat

Transformace dat v Keboole probíhá, jak již bylo výše uvedeno, ve Snowflake SQL, Python a R. Pro správný průběh transformace je důležité nastavit vstupní tabulky v části input mapping, kde je možné zvolit mezi kopírováním a klonováním tabulky. Kopírování tabulky umožňuje data lépe filtrovat (importovat jen část sloupců, nebo například jen část dat), klonování ji nahrává celou, ale velmi rychle (netvoří ji znovu, ale klonuje z úložiště Kebooly pomocí Snowflake SQL funkce Clone).

V rámci transformací je využíváno klonování tabulek, vzhledem k menšímu objemu dat (do milionu řádků u jednotlivých tabulek) totiž filtrování nemá smysl a transformace probíhají rychleji, než kdyby se provádělo kopírování.

V konfiguraci outputu je stěžejní správně nastavit hlavně výstupní bucket a jméno tabulky vytvořené na závěr sekce SQL dotazů, která se zkopíruje do persistentního úložiště a bude dále dostupná.

4.3.6 Zápis transformovaných dat

Zápis dat probíhá v Keboole pomocí writerů, jejichž vytváření je obdobné jako u extraktorů. V současné době je primárně využit zápis .CSV souborů na FTP a zápis přímo do Snowflake databáze, ze které se čerpají data pro reporting.

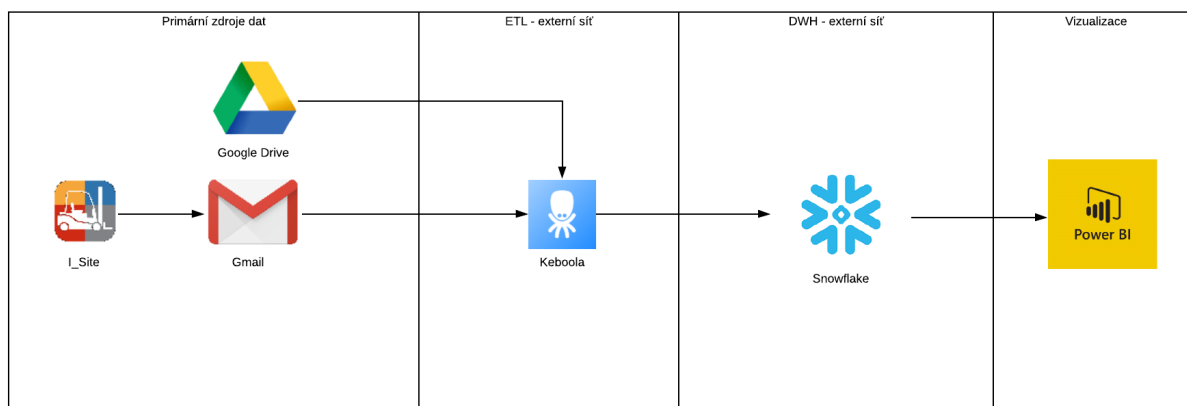
4.3.7 Automatizace kroků ETL procesu

Automatizace kroků, tedy sekvence jednotlivých extraktorů, transformací a writerů, se v Keboole nazývá Orchestrace. Její vytvoření je jednoduché a v rámci jednotlivých fází se vybírají již vytvořené konfigurace extraktorů, případně jednotlivé transformace.

K čemu jsem kritický je problematické spouštění orchestrací v navazujících projektech, což je problém zvláště při využití multiprojektové architektury. Momentálně lze navázat spuštění transformace v jiném projektu na změnu sdílené tabulky (jedna z transformací tak musí být v samostatné fázi, a nevyužívá plně paralelismu), nebo je možné spustit další orchestraci přes POST příkaz vložený do zvláštní Python transformace vložené jen za tímto účelem. Ani jedno řešení není ideální.

4.3.8 Koncový stav toku dat

Datový tok popsáný v rámci této kapitoly se dá shrnout jednoduchým schématem na Obrázku 18.



Obrázek 18: Finální tok dat zpracovávaných v rámci případové studie.

Data jsou odesílána ve formátu .XLS souborů ze systému ISite do Gmailu, malá část dat je přímo vyplňována do Google Tabulek.

ETL nástroj Keboola pomocí API data extrahuje z Google Tabulek a z Gmailu, interně je procesuje a do databáze Snowflake zapisuje tabulky v podobě vhodné pro reporting. Ze Snowflake se potom jednotlivé tabulky propisují do modelu Power BI.

4.3.9 Zhodnocení nástroje Keboola

Cloudová ETL platforma Keboola nabízí velmi dobrý výkon a pohodlné extrahování dat z různých systémů s velmi dobrými podmínkami pro transformování dat. Pro účely tohoto modelového případu funguje perfektně, i při větším objemu dat je extrakce, zpracování i nahrání dat otázkou cca 10 minut.

Případné problémy, které začínají být patrné při delším používání Kebooly pro více různých projektů, se týkají především přehlednosti při správě mnoha datových toků z různých systémů. Aby se byl administrátor i běžný uživatel v Keboola schopen orientovat, je žádoucí využít zmíněnou multiprojektovou architekturu a do ní zavést ještě další jmenné konvence.

V současnosti tak v rámci společnosti jsou u jednotlivých komponent nastaveny následující konvence:

- Extraktory: [Zdroj][Firma][Země][Oblast]
např. [ISite][Firma1][Group][Warehousing]
- Transformace: [PROD/DEV][Firma][Oblast][Charakteristika]
např. [PROD][Firma1][Warehousing][Forklift]
- Writery: [Cíl][Země][Oblast][Charakteristika]
např. [PowerBI][Warehousing][Forklift]

Jako celek hodnotím Keboolu jako velmi povedený nástroj, který zrychluje práci s daty a umožňuje mít uložená data z různých zdrojů na jednom místě, tam je konsolidovat a doplnit o další informace z jiného zdroje nebo transformovat pokročilejší logikou, a zpřístupnit je pro reporting.

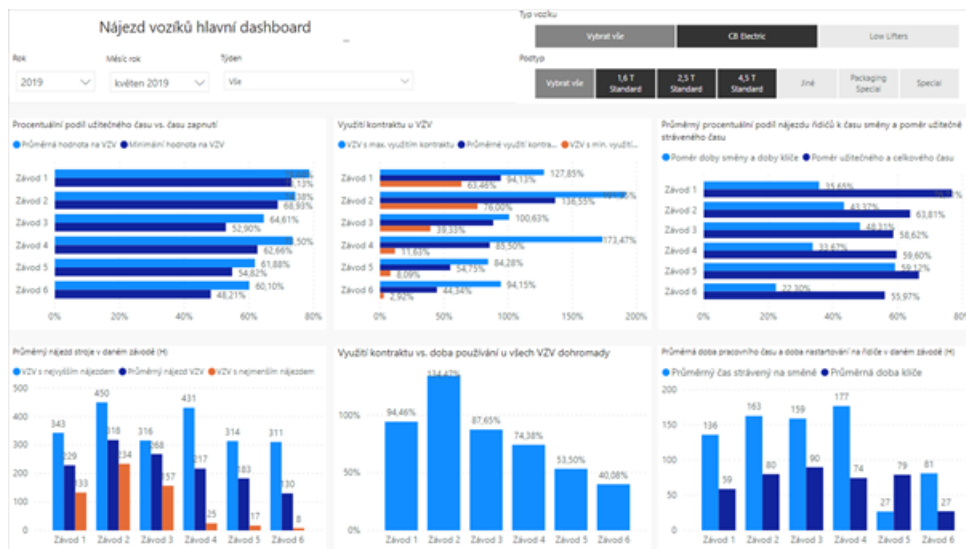
4.4 Vizualizace dat v Power BI

Při tvorbě vizualizace bylo využito softwaru Power BI Desktop v různých měsíčních verzích až po verzi květen 2020. Data ze zdrojového systému byla transformována tak, aby byla co nejlépe připravena pro reporting, nicméně některé transformace probíhají až v Power BI i proto, aby byla demonstrována další funkcionality tohoto nástroje.

4.4.1 Ukázka vybraných vizualizací

V rámci práce je nejdůležitější částí finální vizualizace, od které svou spokojenost nebo nespokojenost s výsledkem odvozují i manažeři. Výsledná podoba první stránky dashboardu je patrná na Obrázku 19, a samotný dashboard je také součástí příloh této práce, stejně jako videoprezentace funkčnosti řešení.

V porovnání s Obrázkem 15 je už nyní vidět, že manažer i při rychlém pohledu na jednotlivé grafy vidí porovnání všech závodů najednou, dokáže rychle odhalit a vytipovat problematické



Obrázek 19: Ukázka hlavní strany dashboardu.

vozíky (maximální vs. minimální využití vozíků stejného nebo obdobného typu). Při vybrání možnosti Detailní analýzy se pak manažer jednoduše přesune na stránku dashboardu, která mu odhalí detailnější pohled např. na detail využití v závodě.



Obrázek 20: Ukázka detailního pohledu na nájezd vozíků v rámci závodu.

To je znázorněno na Obrázku 20. Zde je patrné, že jsou splněny požadavky kladené manažery: v rámci závodu vidí, jak jednotlivé vozíky stojí oproti kontraktu (oranžová čára v grafu) s dobou využití. Jako zajímavost je uvedena i doba nečinnosti, nicméně vzhledem ke kontraktu počítaném jen na dobu skutečného používání je z pohledu využití vozíku zajímavé reálné užívání.

Poměr skutečně využitého času a celkové doby zapnutí stroje je potom zajímavý spíše z pohledu řidičů, nebo při porovnání vytížení vozíků, pro jehož vizualizaci byl zvolen Ganttův diagram – „plná“ část znázorňuje procentuálně užitečnou dobu, světlá část procentuální dobu nečinnosti vozíku.

I z těchto heslovitých ukázek je zřejmé, že mnohem větší informační hodnotu mají přehledně graficky zorganizované dashboardy, v nichž je člověk v souladu s teorií schopen odhalit vzorce a další zákonitosti lépe než v rozsáhlé tabulce, jako tomu bylo v původním reportingu v nástroji Toyota ISite.

4.4.2 Datový model

Do datového modelu byla importována data z databáze Snowflake, kvůli publikaci bakalářské práce byla v příložené verzi tato napojení nahrazena napojením na prosté .CSV soubory. Tyto jsou dostupné v příloze, stejně jako celý datový model v Power BI.

Nejdůležitějšími body v datovém modelu Power BI bylo nahrávat tabulky s daty tak, aby byly optimalizovány pro reporting. To znamená, nejlépe připravit hvězdicové schéma, s tím, že tabulek s transakčními daty je více, a dimenzionální tabulky jsou tak propojeny na několik tabulek s transakčními daty viz schéma na Obrázku 21. Model je ve hvězdicový, částečně denormalizovaný.

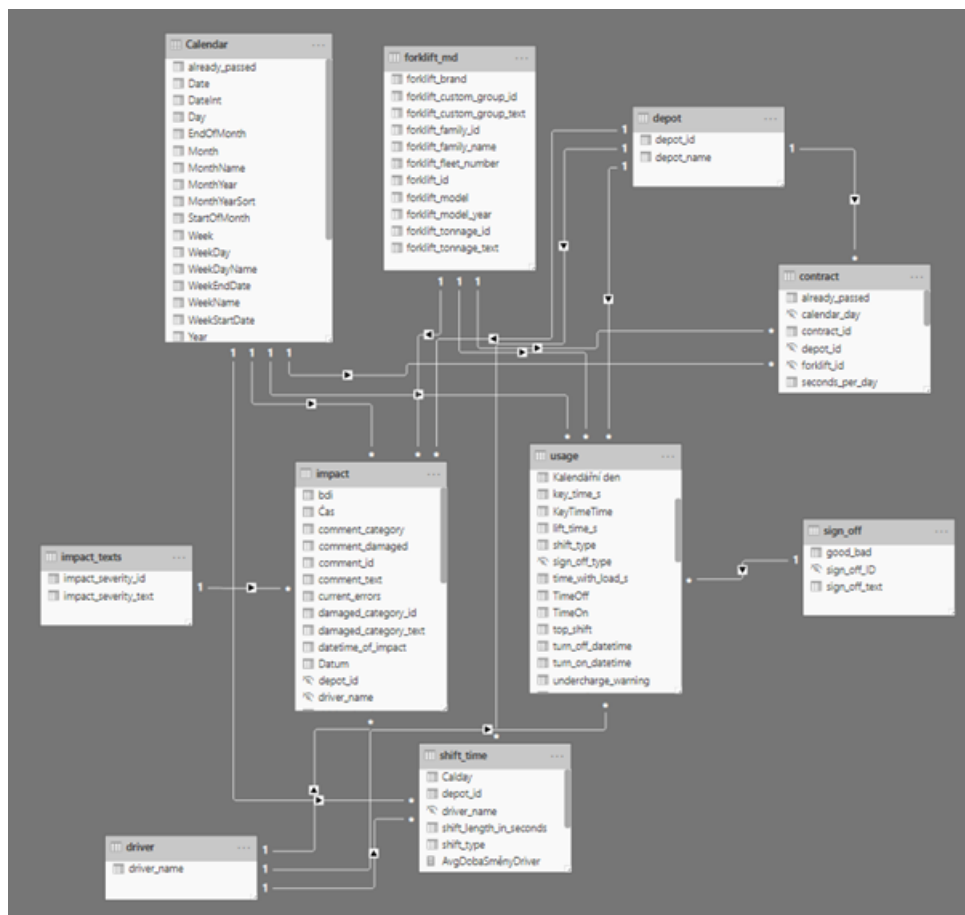
Přestože jsem v porovnání nástrojů psal o nevýhodě, kterou Power BI má, a to je možnost propojit tabulky jen skrze jedno pole, v tomto konkrétním případě nešlo o problém, který by se musel řešit (standardním řešením je spojení polí primárního klíče do jednoho textového řetězce a následný join přes toto umělé pole).

Jako velmi důležité se při práci samotné ukázalo v rámci transakčních dat skrývat jednotlivé charakteristiky, které mají svůj obraz i v tabulkách master dat, tedy jednotlivých dimenzí. V rámci vizualizace je totiž důležité filtrovat nad těmito společnými dimenzemi a využívat je na správných osách propojených grafů. Omylem použitá charakteristika pocházející z tabulky transakčních dat pak ústí v nefunkční vizualizace, či ve vizualizace se zkreslenými výsledky.

4.4.3 Dopočítávání tabulek a sloupců v Power BI

Z teoretických zdrojů i z praktické zkušenosti je kvůli výkonu při datovém modelování upřednostňováno dělat co nejvíce práce při transformaci dat na straně zdroje, resp. ETL nástroje. Až pokud toto nejde, nebo to není efektivní, lze se přesunout k úpravě dat v rámci nahrávání do aplikace Power BI (oblast transformace dat). Zde je v případě Power BI využíván jazyk M, který umožňuje např. na základě jednoho sloupce přidat sloupec další, nebo i vytvořit zcela novou tabulku. Jako třetí možnost je pak využití jazyka DAX, jenž je podobný výrazům v tabulkovém procesoru Excel, a jež rovněž umožňuje dopočítání některých charakteristik.

Vzhledem k tomu, že cílem práce bylo vyzkoušet možnosti i tohoto nástroje, vytvořil jsem některou logiku přímo nad daty v Power BI, aby bylo možno otestovat i tyto způsoby tvorby tabulek. Musím říct, že výrazného zpomalení nebo velkého zatížení systému jsem si při využívání



Obrázek 21: Datový model dashboardu v Power BI.

těchto polí a tabulek nevšiml: nicméně v případech, které jsem dopočítával, jde o nízké stovky tisíc záznamů, což pro moderní stroje není příliš zatěžující.

Jako dobrý příklad dimenzionální tabulky vhodné (a nutné) k provázání různých transakčních dat, je dopočítaná kalendářní tabulka. Ta je vytvořená jednoduchým DAX zápisem, resp. funkcí Calendar, do které vstupují jako proměnné datумы počátečního data tvořeného kalendáře a data koncového. V tomto konkrétním případě Výpisu 2 bylo zvoleno rozmezí od 01.01.2019 do 31.12.2022.

1 `Calendar = CALENDAR(DATE(2019;01;01); DATE(2022;12;31))`

Výpis 2: Vytvoření datové tabulky v jazyce DAX.

Tato datová tabulka je pak skrz DAX dále rozšiřována o dopočtená pole pro každý řádek tabulky představující jedno konkrétní datum. Do datové věty je vhodné přidat další sloupce jako je například informace, který měsíc je který den, jaký týden spadá, kdy daný měsíc nebo týden začíná a končí a další. V rámci tabulky lze udělat také pole, které slouží pro filtrování dat v případě, že dané datum ještě nenastalo. Lze tak ošetřit třeba počítání poměru mezi reálným

využitím a kontraktem ve chvíli, kdy je reálně polovina měsíce (a realitu chceme srovnat s plánem, který máme po dnech) – lze tak započítat hodnoty plánu jen pro ty dny, které již uběhly.

Další zajímavou dopočtenou tabulkou je shifttime vygenerovaná dle výpisu 3, kde se dopočítává délka směny pro řidiče podle toho, jestli měli v ranní, odpolední nebo noční směně zapnutý nějaký stroj. Pokud ano, automaticky do tabulky přibude záznam, v němž bude figurovat daný řidič s délkou směny v sekundách.

```
1 shifttime =  
2 SUMMARIZE('usage';  
3 'usage'[Calday];  
4 'usage'[driver_name];  
5 'usage'[depot_id];  
6 'usage'[shift_type];  
7 "shiftlengthinseconds"; 28880)
```

Výpis 3: Vytvoření tabulky s odhadovými směny řidičů v jazyce DAX.

Využití M a DAXu k dopočtení sloupců, které nebyly obsahem transakčních nebo master dat je z praktického pohledu jednoduché a odemyká pro tvůrce vizualizací a analytiky další možnosti, jak jednoduše vizualizovat data. Také mohou sloužit jako první rychlý návrh nové tabulky či oblasti dat k využití v nové vizualizaci a lze je začlenit do datového modelu až poté, co se prokáže praktická užitečnost nové vizualizace.

4.4.4 Problémy s datovým modelováním

V průběhu zpracování práce se objevilo několik problémů souvisejících s datovými modely. Tím prvním a pro mě nejvíce překvapivým byla nevyhovující práce s datovým typem DATETIME. Pro využití ve filtrech, a i ve vizualizacích bylo nakonec mnohem jednodušší pracovat s více sloupci, kde bylo zvláště uvedeno datum a zvláště čas, samozřejmě ve formátech DATE a TIME.

Vzhledem k tomu bylo i v transformačním procesu provedeno rozdělení jízdy, která začala před půlnocí a skončila po půlnoci druhého dne do dvou řádků – tak, aby každá jízda skončila před půlnocí a začala po půlnoci. Jednotlivé statistiky využití vozíku se pak dělí poměrově.

Konkrétně v této práci v rámci exportu dat z Kebooly a jejich importu do Power BI sice importuji raději DATETIME, ale přímo v Power BI potom využívám rozdělení sloupců na datum a čas začátku a konce. Bez tohoto rozdělení totiž vizualizace nefungovala. Jako další krok ke zlepšení výkonu by bylo záhodné upravit datový model již v nástroji Keboola.

4.4.5 Využití metrik

Při práci se složitějšími a obsáhlejšími modely je zřejmé, že na reporting jakýchkoliv číselných výsledků je důležité použít metriky. PowerBI tento styl práce nevynucuje, na rozdíl např. od

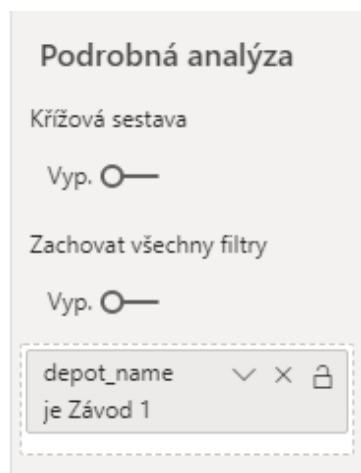
QlikSense, ale při delším používání je zřejmé, že právě správné sestavení metrik je klíčové pro správnou interpretaci vizualizací a pro možnost opakovaného použití stejného vzorce při pohledu na data z různé úrovně agregace.

V rámci práce bylo nejčastěji využíváno sumací pomocí funkce SUM nad všemi zafiltrovanými daty, průměrování dle jednotlivých VZV a řidičů za využití funkce AVG a také přepočítávání v rámci časové délky trvání ze sekund na hodiny. Speciálním případem pro zobrazování dat v cílovém formátu a maximálním detailu byl přepočet sekund do formátu hodin, minut a sekund a jeho následné zobrazení pomocí textového řetězce. Formát TIME by bohužel při hodnotě vyšší než 24 hodin začal počítat opět od půlnoci.

4.4.6 Vytváření propojených dashboardů a vizualizací

V rámci případové studie bylo vytvořeno několik dashboardů s různými vizualizacemi, které umožňují propojení na detailnější pohled. V případě Power BI jsou pro snadnou navigaci mezi jednotlivými dashboardy a zachování filtrů pohledu důležitá dvě nastavení: nastavení tzv. Podrobné analýzy (Drilldown), nastavení synchronizace Průřezů (Slicer) a nastavení interakce Průřezů s vizualizacemi.

Podrobná analýza umožňuje u cílového dashboardu nastavit, z jaké charakteristiky je možné se na konkrétní report prokliknout. Nenápadné pod-nastavení „zachovat všechny filtry“ je nicméně klíčové pro správnou funkci reportingu, viz Obrázek 22. Pokud je toto nastavení zatrženo, aplikují se na daný pohled všechny filtry ze zdrojové vizualizace a z detailu prvku dané vizualizace.



Obrázek 22: Nastavení podrobné analýzy v Power BI.

Pokud tak např. v případě reportingu intenzity nárazů kliknete na podrobnou analýzu těch s nízkou intenzitou, Power BI automaticky zafiltruje nízkou intenzitu nárazu, závod a vše vybrané v dalších průřezích. Na stránce detailu závodu tak již není možné dále upravovat detail filtrování, což není žádoucí.

Řešením je vypnutí nastavení zachovat všechny filtry a synchronizace Průřezů napříč dashboardy, která se dělá skrze menu Zobrazení a položku Synchronizovat průřezy 23. Zde si pak lze vybrat, které průřezy mají být synchronizovány napříč jednotlivými stránkami.



Obrázek 23: Nastavení synchronizace průřezů.

4.4.7 Zhodnocení tvorby případové studie

V rámci případové studie byly provedeny všechny kroky vytvoření funkční vizualizace od úvodní extrakce dat, jejich transformace a nahrání do reportovací databáze. K této extrakci výborně posloužila cloudová ETL platforma Keboola.

Samotná vizualizace pak proběhla ve zvoleném nástroji PowerBI. V kapitole 4.4. pak byla shrnuta specifika některých vizualizačních kroků a principů. Veškeré konfigurace pro ETL v rámci platformy Keboola, transformace dat a data, na nichž byl vystavěn report včetně samotné vizualizace jsou k dispozici v příloze této práce. Jedinou změnou oproti produkčnímu nasazení ve firmě je přepojení na zdroj dat – místo Snowflake jsou využity lokální .csv soubory.

Vizualizace má velmi rychlou odezvu, dle uživatelů se v ní dobře orientuje a přináší potřebné informace formou interaktivních grafů.

5 Shrnutí praktických výsledků

V rámci práce byla provedena analýza tří moderních analyticko-vizualizačních nástrojů. Na základě tohoto porovnání se spolupracující společnost rozhodla implementovat doporučený software Power BI a v současné době probíhá budování nového reportingu právě na této platformě.

Zkušenosti ve společnosti jsou zatím velmi pozitivní, uživatelé dokáží Power BI rychle využít ať už jako vizualizační nástroj na sledování stanovených ukazatelů, anebo jako zdroj dat pro další analýzu v Excelu. Pozitivně je také hodnoceno propojení se Snowflake a Keboola, kdy rozšiřování dat o externí vstupy je mnohem rychlejší a jednodušší než v rámci dříve používaného SAP BW.

Konkrétní vypracování modelového příkladu v rámci případové studie pak mělo velmi pozitivní ohlas při hodnocení jednotlivých řidičů vysokozdvížných vozíků a benchmarkingu napříč závody. Kvitována byla i vizualizace formou Ganttova diagramu ve chvíli, kdy se jednalo o dokoupení jednoho speciálního VZV - na vizualizaci bylo možno lehce ukázat, že daný VZV je potřeba pro optimální rozložení práce v rámci skladu. Po zavedení jednoduchého monitorování v Power BI také klesl počet chybných odhlášení od strojů.

Uživatelské ohlasy na report jsou velmi pozitivní - hlavní benefity Power BI na tomto konkrétním reportu představují i v přiloženém videu v příloze této bakalářské práce.

6 Závěr

Práce je zaměřena především na vizualizační část práce s BI nástroji. V rámci teoretické části je objasněna základní terminologie Business Intelligence a základní proces vytváření vizualizace, přičemž tyto pojmy jsou dále využívány v praktické části práce.

Jádrem práce je srovnání vizualizačních nástrojů Power BI, Tableau a Qlik Sense z hlediska možnosti vizualizace dat v předpřipravených datových modelech. Hodnocena je především jednoduchost návrhu těchto vizualizací, rychlost nahrávání dat, možnost dalšího obohacování datových modelů a spojování důležitých tabulek, uživatelská přívětivost. Ve srovnání je také zohledněn fakt, zda se daný nástroj hodí pro nasazení v korporátní společnosti zejména z pohledu ceny a z pohledu toho, jak bude daný nástroj využíván. V celkovém hodnocení je jako nejvhodnější nástroj vybráno Power BI, především kvůli ceně a jednoduchosti nasazení.

Tato možnost srovnání tří alternativ vizualizačního softwaru byla pro autora klíčová pro orientaci na trhu moderních BI nástrojů. Bylo skutečně poučné si komfort a styl práce v jednotlivých nástrojích vyzkoušet, neboť v marketingových materiálech je samozřejmě každý nástroj líčen jako bezproblémový a vhodný pro všechny příležitosti. I když základní reporting pro firmu lze sestavit v každém z nich, je nutné mít na paměti všechny jejich silné i slabé stránky.

V rámci případové studie je pak na příkladu reportingu vysokozdvížných vozíků ukázán princip toku dat ze zdrojové databáze do databáze reportingové a konkrétní doporučené nastavení platformy Keboola. Tuto část hodnotím jako užitečnou z hlediska naučení se principům datového modelování na ETL platformě, a toku dat.

Následně jsou popsána specifika vizualizace v Power BI včetně konečného návrhu vizualizace využití vysokozdvížných vozíků. Vytvořené vizualizace mohou být užitečné pro další analytiky, kteří by si rádi vyzkoušeli možnosti vizualizačních nástrojů a také k demonstraci funkčnosti Power BI.

V neposlední řadě je nastavení ETL, oblast datového skladu i dashboard se všemi vizualizacemi součástí standardizovaného reportingu podniku a v praxi je používán k identifikaci možných prostožů, rizikových často bourajících řidičů, nebo úspor v rámci využití vysokozdvížných vozíků.

Konkrétní benefity interaktivní vizualizace jsou opětovně shrnuty ve videu jež je součástí přílohy této práce, stejně jako je součástí hotová vizualizace ve formátu PBIX.

Literatura

1. WURMAN, Richard Saul. *Information Anxiety*. Doubleday, 1989. ISBN 978-0385243940.
2. CUESTA, Hector. *Analýza dat v praxi*. Computer Press, 2015. ISBN 978-8025143612.
3. *What is business intelligence? Your guide to BI and why it matters* [online]. Tableau Software [cit. 2020-04-30]. Dostupné z: <https://www.tableau.com/learn/articles/business-intelligence>.
4. NOVOTNÝ, Ota; POUR, Jan; SLÁNSKÝ, David. *Business Intelligence: Jak využít bohatství ve vašich datech*. Grada Publishing, 2005. ISBN 80-24710943.
5. YAU, Natahan. *Visualize This*. Wiley Publishing, Inc., 2011. ISBN 978-0470944882.
6. FRY, Ben. *Visualizing Data*. O'Reilly Media, 2008. ISBN 978-0596514556.
7. SHERMAN, Rick. *Business Intelligence Guidebook: From Data Integration to Analytics*. Morgan Kaufmann, 2014. ISBN 978-0124114616.
8. REDDY, Dinesh Kumar; MANOKAR, Manjari. *How Data Integration works in Snowflake?* [Online]. Visual BI Solutions [cit. 2020-05-10]. Dostupné z: <https://visualbi.com/blogs/snowflake/data-integration-works-snowflake/>.
9. POUR, Jan; MARYŠKA, Miloš; STANOVSKÁ, Iva; ŠEDIVÁ, Zuzana. *Self Service Business Intelligence*. Grada Publishing, a.s., 2018. ISBN 978-8027106165.
10. DECKLER, Greg. *Learn Power BI: A beginner's guide to developing interactive business intelligence solutions using Microsoft Power BI*. Morgan Kaufmann, 2014. ISBN 978-0124114616.
11. *Star and Snowflake Schemas* [online]. Oracle [cit. 2020-05-15]. Dostupné z: https://www.oracle.com/webfolder/technetwork/tutorials/obe/db/10g/r2/owb/owb10gr2_gs/owb/lesson3/starandsnowflake.htm.
12. MILLIGAN, Joshua N. *Learning Tableau 2019: Tools for Business Intelligence, data prep and visual analytics*. Packt Publishing, 2019. ISBN 978-1788839525.
13. *2019 Gartner Magic Quadrant for Analytics and Business Intelligence Platforms* [online]. Qlik [cit. 2019-12-10]. Dostupné z: <https://www.qlik.com/us/gartner-magic-quadrant-business-intelligence>.
14. *Ceny Power BI: Analýzy pro libovolnou organizaci* [online]. Microsoft [cit. 2020-05-12]. Dostupné z: <https://powerbi.microsoft.com/cs-cz/pricing/>.
15. *Pricing for data people: Teams and Organizations* [online]. Tableau [cit. 2020-05-12]. Dostupné z: <https://www.tableau.com/pricing/teams-orgs>.
16. *Qlik Pricing* [online]. Qlik [cit. 2020-05-12]. Dostupné z: <https://www.qlik.com/us/pricing>.

17. *ISite - Management flotily* [online]. Toyota [cit. 2020-05-05]. Dostupné z: https://toyota-forklifts.cz/reseni/i_site-fleet-management/.
18. *Processors* [online]. Keboola [cit. 2020-05-05]. Dostupné z: <https://developers.keboola.com/extend/component/processors/#using-processors-with-configuration-rows>.
19. *Keboola Component Configurations: Get Components* [online]. Keboola [cit. 2020-05-05]. Dostupné z: <https://keboola.docs.apiary.io/#reference/component-configurations/get-components>.

A Struktura složek v rámci přílohy

/ -> základní složka přílohy

- ukazka_keboola -> ukázka práce v cloudové ETL platformě Keboola
 - 00 -> Extrakce a základní datová transformace
 - * contract -> složka s daty a zdrojovými kódy kontraktu VZV
 - * forklift -> složka s daty a zdrojovými kódy kmenových dat VZV
 - * impact -> složka s daty a zdrojovými kódy nárazů s VZV
 - * usage -> složka s daty a zdrojovými kódy využití VZV
 - 10 -> Transformace a doplnění business logiky
 - * contract -> složka s daty a zdrojovými kódy kontraktu VZV
 - * forklift -> složka s daty a zdrojovými kódy kmenových dat VZV
 - * impact -> složka s daty a zdrojovými kódy nárazů s VZV
 - * usage -> složka s daty a zdrojovými kódy využití VZV
 - 15 -> Datový sklad
 - * contract -> složka s daty a zdrojovými kódy kontraktu VZV
 - * forklift -> složka s daty a zdrojovými kódy kmenových dat VZV
 - * impact -> složka s daty a zdrojovými kódy nárazů s VZV
 - * usage -> složka s daty a zdrojovými kódy využití VZV
- ukazka_powerbi -> ukázka vizualizační části práce
 - Forklifts.pbix -> PBIX soubor s přichystaným interaktivním dashbordem, který lze otevřít a prohlédnout v Microsoft Power BI Desktop
 - PBI_VZV_prezentace.mkv -> videoprezentace řešení v Power BI

B Orientace v příložené ukázce ETL Keboola

Ve složce ukazka_keboola jsou k dispozici zdrojová data a průběh transformací na cloudové ETL platformě Keboola.

Rozdělení složek 00, 10 a 15 odpovídá popsané projektové struktuře v kapitole 4.3.3 této práce.

Struktura se dále dělí na jednotlivé podsložky – logické celky, s nimiž se dále pracuje. Tok dat napříč projekty vysvětlím na jednoduchém příkladu Kontraktu. Ve složce ukazka_keboola/00/contract jsou následující podsložky:

- Extraction -> ukázka extrakce dat.
 - configuration.json -> konfigurace extraktoru Gmail příloh v Keboole s technickým názvem apac.ex-gmail-attachments.
 - processors.json -> procesory použité po extrakci.
 - ukazka_toyota_kontrakt.xlsx -> ukázkový Excel se smazanými citlivými daty, který se pravidelně exportuje ze systému ISite.
- Transformation -> úvodní transformace dat.
 - contract_transformation.sql -> transformace v jazyku SQL.
 - input_output_mapping_contract.png -> vstupní/výstupní mapování.
 - input_contract.csv -> vstupní soubor do transformace/výstupní soubor extrakce.

Ve složce ukazka_keboola/10/contract je již pouze podsložka transformation s následujícím obsahem:

- contract_transformation.sql -> transformace v jazyku SQL .
- input_output_mapping_contract.png -> vstupní/výstupní mapování.
- input_contract.csv -> vstupní soubor, jež je zároveň výstupem transformace z 00.

Ve složce ukazka_keboola/15 jsou dvě podsložky:

- input_data -> výstupní data z 10, která slouží jak vstup 15 a odtud se zapisují do Snowflake - jde o zdroje vizualizace v Power BI.
 - contract.csv -> zdrojový soubor délky kontraktů použitý pro vytvoření vizualizace, vytvořený transformací 10.
 - driver.csv -> zdrojový soubor id řidičů použitý pro vytvoření vizualizace, vytvořený transformací v 10 (transformace usage).

- forklift.csv -> zdrojový soubor kmenových dat vozíků použitý pro vytvoření vizualizace, vytvořený transformací v 10.
- impact.csv -> zdrojový soubor nárazů s vozíky použitý pro vytvoření vizualizace, vytvořený transformací v 10.
- usage.csv -> zdrojový soubor jízd vozíků použitý pro vytvoření vizualizace, vytvořený transformací v 10.
- writer_setup -> nastavení writeru v Keboola do Snowflake databáze.
 - db_credentials_writer.png -> nastavení přihlášení k Snowflake.
 - writer_setup.png -> nastavení, které tabulky má Keboola do DWH Snowflake zapsat.